Practical File

Q1. 1. Display the pattern:

1

1 2

1 2 3

Take ‘n’ in a textbox from user. Display this pattern using

* Scriptlets

.jsp file

<html>

<body>

<% int n=Integer.parseInt(request.getParameter("t1"));

for(int i=1;i<=n;i++)

{ for(int j=1;j<=i;j++)

{ %> <%=j%>&nbsp&nbsp

<%}%><br><%

}

%>

</body>

</html>

.html file

<html>

<body>

<form action="number.jsp">

Enter any no.<br>

<input type="text" name="t1"><br>

<input type="submit">

</form>

</body>

</html>

* <c:forEach> loop

.jsp file

<%@ page contentType="text/html" %>

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

<html>

<body>

<c:set var="n" value="${param.t1}"/>

<c:forEach var="i" begin="1" end="${n}" >

<c:forEach var="j" begin="1" end="${i}">

${j}

</c:forEach>

<br>

</c:forEach>

</body>

</html>

.html file

<html>

<body>

<form action="num.jsp">

Enter any no.<br>

<input type="text" name="t1"><br>

<input type="submit">

</form>

</body>

</html>

Q2. Make two files as follows:

a. main.html: shows 2 text boxes and 3 radio buttons with values "addition",

"subtraction" and "multiplication"

.jsp file

<html>

<body>

<% int n1=Integer.parseInt(request.getParameter("t1"));

int n2=Integer.parseInt(request.getParameter("t2"));

String r=""+request.getParameter("r1");

if(r.equals("1"))

{%> Result is:&nbsp<%=n1+n2%>

<%}

if(r.equals("2"))

{%> Result is:&nbsp<%=n1-n2%>

<%}

if(r.equals("3"))

{%> Result is:&nbsp<%=n1\*n2%>

<html>

<body>

<form action="operate.jsp">

Enter first no.<br>

<input type="text" name="t1"><br>

<br>Enter second no.<br>

<input type="text" name="t2"><br>

<br><input type="radio" name="r1" value="1">Addition<br>

<input type="radio" name="r1" value="2">Subtraction<br>

<input type="radio" name="r1" value="3">Multiplication<br>

<input type="submit">

</form>

</body>

</html> <%}

%>

</body>

</html>

.html file

<html>

<body>

<form action="operate.jsp">

Enter first no.<br>

<input type="text" name="t1"><br>

<br>Enter second no.<br>

<input type="text" name="t2"><br>

<br><input type="radio" name="r1" value="1">Addition<br>

<input type="radio" name="r1" value="2">Subtraction<br>

<input type="radio" name="r1" value="3">Multiplication<br>

<input type="submit">

</form>

</body>

</html>

b. operate.jsp: depending on what the user selects perform the corresponding function

(Give two implementations: using request.getParameter() and using expression

language)

.html

<html>

<body>

<form action="op.jsp">

Enter first no.<br>

<input type="text" name="t1"><br>

<br>Enter second no.<br>

<input type="text" name="t2"><br>

<br><input type="radio" name="r1" value="1">Addition<br>

<input type="radio" name="r1" value="2">Subtraction<br>

<input type="radio" name="r1" value="3">Multiplication<br>

<input type="submit">

</form>

</body>

</html>

.jsp

<%@ page contentType="text/html" %>

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

<html>

<body>

<c:choose>

<c:when test="${param.r1 == '1'}">

Result is: &nbsp ${param.t1+param.t2}

</c:when>

<c:when test="${param.r1 == '2'}">

Result is: &nbsp ${param.t1-param.t2}

</c:when>

<c:when test="${param.r1 == '3'}">

Result is: &nbsp ${param.t1\*param.t2}

</c:when>

<c:otherwise>

Please select appropriate choice.

</c:otherwise>

</c:choose>

</body>

</html>

Q3. Validate User input entered in a form. The input must include Name, DOB, Email ID,Lucky Number, Favorite food etc.

<%@ page contentType="text/html" %>

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

<html>

<head>

<title>User Info Entry Form</title>

</head>

<body bgcolor="white">

<form action="validate.jsp" method="post">

<input type="hidden" name="submitted" value="true">

<table>

<c:if test="${param.submitted && empty param.userName}">

<tr><td></td>

<td colspan="2"><font color="red">

Please enter your Name

</font></td></tr>

</c:if>

<tr>

<td>Name:</td>

<td>

<input type="text" name="userName" value="<c:out value="${param.userName}" />">

</td>

</tr>

<c:if test="${param.submitted && empty param.birthDate}">

<tr>

<td></td>

<td colspan="2"><font color="red">

Please enter your Birth Date

</font></td></tr>

</c:if>

<tr>

<td>Birth Date:</td>

<td>

<input type="text" name="birthDate" value="<c:out value="${param.birthDate}" />">

</td>

<td>(Use format yyyy-mm-dd)</td>

</tr>

<c:if test="${param.submitted && empty param.emailAddr}">

<tr><td></td>

<td colspan="2"><font color="red">

Please enter your Email Address

</font></td></tr>

</c:if>

<tr>

<td>Email Address:</td>

<td>

<input type="text" name="emailAddr" value="<c:out value="${param.emailAddr}" />">

</td>

<td>(Use format name@company.com)</td>

</tr>

<c:if test="${param.submitted && param.gender != 'm' && param.gender != 'f'}">

<tr><td></td>

<td colspan="2"><font color="red">

Please select a valid Gender

</font></td></tr>

</c:if>

<tr>

<td>Gender:</td>

<td>

<c:choose>

<c:when test="${param.gender == 'f'}">

<input type="radio" name="gender" value="m">

Male<br>

<input type="radio" name="gender" value="f" checked>

Female

</c:when>

<c:otherwise>

<input type="radio" name="gender" value="m" checked>

Male<br>

<input type="radio" name="gender" value="f">

Female

</c:otherwise>

</c:choose>

</td>

</tr>

<c:if test="${param.submitted && (param.luckyNumber < 1 || param.luckyNumber > 100)}">

<tr><td></td>

<td colspan="2"><font color="red">

Please enter a Lucky Number between 1 and 100

</font></td></tr>

</c:if>

<tr>

<td>Lucky number:</td>

<td>

<input type="text" name="luckyNumber" value="<c:out value="${param.luckyNumber}" />">

</td>

<td>(A number between 1 and 100)</td>

</tr>

<c:forEach items="${paramValues.food}" var="current">

<c:choose>

<c:when test="${current == 'z'}">

<c:set var="pizzaSelected" value="true" />

</c:when>

<c:when test="${current == 'p'}">

<c:set var="pastaSelected" value="true" />

</c:when>

<c:when test="${current == 'c'}">

<c:set var="chineseSelected" value="true" />

</c:when>

<c:otherwise>

<c:set var="invalidSelection" value="true" />

</c:otherwise>

</c:choose>

</c:forEach>

<c:if test="${invalidSelection}">

<tr><td></td>

<td colspan="2"><font color="red">

Please select only valid Favorite Foods

</font></td></tr>

</c:if>

<tr>

<td>Favorite Foods:</td>

<td>

<input type="checkbox" name="food" value="z"

${pizzaSelected ? 'checked' : ''}>Pizza<br>

<input type="checkbox" name="food" value="p"

${pastaSelected ? 'checked' : ''}>Pasta<br>

<input type="checkbox" name="food" value="c" ${chineseSelected ? 'checked' : ''}>Chinese

</td>

</tr>

<tr>

<td colspan="3">

<input type="submit" value="Send Data">

</td>

</tr>

</table>

</form>

</body>

</html>

Q4. Display Good Morning <uname>, Good Afternoon <uname> or Good Evening

<uname> based on the current time of the day.

.html

<html>

<body>

<form action="operate.jsp">

Enter first no.<br>

<input type="text" name="t1"><br>

<br>Enter second no.<br>

<input type="text" name="t2"><br>

<br><input type="radio" name="r1" value="1">Addition<br>

<input type="radio" name="r1" value="2">Subtraction<br>

<input type="radio" name="r1" value="3">Multiplication<br>

<input type="submit">

</form>

</body>

</html>

.jsp

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

<html>

<body bgcolor="white">

<jsp:useBean id="clock" class="java.util.Date" />

<c:choose>

<c:when test="${clock.hours < 12}">

<h1>Good morning!</h1>

</c:when>

<c:when test="${clock.hours < 18}">

<h1>Good afternoon! </h1>

</c:when>

<c:otherwise>

<h1>Good evening!</h1>

</c:otherwise>

</c:choose>

Welcome to our site, open 24 hours a day.

</body>

</html>

Q5. Let the user enter a word a in a textbox and let her/him select one of even or odd radio buttons. If she/he selects odd, check the odd positions in the word entered, if they all contain vowels, then display the message ‘You win’, else display ‘You lose’. Similarly, if the user selects even, check for vowels in all even positions in the word entered. Use jstl’s ‘fn’ library.

.html

<html>

<body>

<form action="process.jsp">

Enter any String<br>

<input type="text" name="t1"><br>

Select any one:

<input type="radio" name="n" value="odd">Odd<br>

<input type="radio" name="n" value="even">Even<br>

<input type="submit">

</form>

</body>

</html>

.jsp

<%@ page contentType="text/html" %>

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

<%@ taglib prefix="fn" uri="http://java.sun.com/jsp/jstl/functions" %>

<html>

<body>

<c:set var="word" value="${param.t1}"/>

<c:set var="mode" value="${param.n}"/>

<c:if test="${n="odd"}">

<c:set var="i" value="1"/>

</c:if>

<c:set var="mode" value="${param.n}"/>

<c:if test="${n="even"}">

<c:set var="i" value="0"/>

</c:if>

<c:forEach var="x" begin="${i}" end="${fn:length(word)}" step="2" >

<c:set var="ch" value="${fn:substring(word,x,x)}"/>

<c:if test="${!fn:containsIgnoreCase("aeiou",ch)}">

<c:set var="win" value="false"/>

</c:if>

</c:forEach>

<c:set var="win" value="true"/>

<c:choose>

<c:when test="${win==true}">

<h1>YOU WIN</h1>

</c:when>

<c:otherwise>

<h1>YOU LOSE</h1>

</c:otherwise>

</c:choose>

</body>

</html>

Q6. Create your custom library which contains two tags: <hello>, <choco>.

Usage of the tags:

* <hello name=”Ajay”>: Output should be Hello Ajay. It contains a mandatory attribute ‘name’ which can accept Dynamic value.

.jsp

<%@ taglib uri="my" prefix="MyTag" %>

<html>

<body>

<MyTag:hello name="User" />

</body>

</html>

.java

package my;

import javax.servlet.\*;

import javax.servlet.http.\*;

import javax.servlet.jsp.\*;

import javax.servlet.jsp.tagext.TagSupport;

import java.io.\*;

import java.util.\*;

public class first extends TagSupport

{ JspWriter out;

private String name;

public void setName(String name)

{ this.name=name;

}

public int doStartTag()throws JspException

{ out=pageContext.getOut();

return SKIP\_BODY;

}

public int doEndTag()throws JspException

{

try{ out.println(" Hello "+name);

}

catch(Exception e)

{

try{

out.println(e);

}

catch(Exception ec)

{ ec.printStackTrace();

}

}

return EVAL\_PAGE;

}

}

.tld

<?xml version="1.0" encoding="ISO-8859-1" ?>

<!DOCTYPE taglib

PUBLIC "-//Sun Microsystems, Inc.//DTD JSP Tag Library 1.2//EN"

"http://java.sun.com/j2ee/dtd/web-jsptaglibrary\_1\_2.dtd">

<taglib>

<tlib-version>1.0</tlib-version>

<jsp-version>1.2</jsp-version>

<short-name>MyTag</short-name>

<uri>my</uri>

<tag>

<name>hello</name>

<tag-class>my.first</tag-class>

<body-content>empty</body-content>

<attribute>

<name>name</name>

<type>java.lang.String</type>

<required>true</required>

<rtexprvalue>true</rtexprvalue>

</attribute>

</tag>

</taglib>

Web.xml

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

<web-app version="2.5"

xmlns="http://java.sun.com/xml/ns/javaee"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://java.sun.com/xml/ns/javaee

http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd">

<jsp-config>

<taglib>

<taglib-uri>my</taglib-uri>

<taglib-location>/WEB-INF/tlds/mydate.tld</taglib-location>

</taglib>

</jsp-config>

</web-app>

* <choco texture=”Chewy”>: Output should be FiveStar, BarOne.<choco texture=”Crunchy”>: Output should be Munch. KitKat. That means the mandatory attribute must accept a value, and based on the attributes value, it should give output. You must use a bean ChocoBean for this purpose.

.jsp

<%@ taglib uri="my" prefix="MyTag" %>

<html>

<body>

<MyTag:choco texture="Chewy" />

</body>

</html>

.java

package my;

import javax.servlet.\*;

import javax.servlet.http.\*;

import javax.servlet.jsp.\*;

import javax.servlet.jsp.tagext.TagSupport;

import java.io.\*;

import java.util.\*;

public class first extends TagSupport

{ JspWriter out;

private String texture;

public void setTexture(String texture)

{ this.texture=texture;

}

public int doStartTag()throws JspException

{ out=pageContext.getOut();

return SKIP\_BODY;

}

public int doEndTag()throws JspException

{

try{ if(texture.equalsIgnoreCase("Chewy"))

{out.println("FiveStar, BarOne");

}

if(texture.equalsIgnoreCase("Crunchy"))

{out.println("Munch, Kitkat");

}

}

catch(Exception e)

{

try{

out.println(e);

}

catch(Exception ec)

{ ec.printStackTrace();

}

}

return EVAL\_PAGE;

}

}

.tld

<?xml version="1.0" encoding="ISO-8859-1" ?>

<!DOCTYPE taglib

PUBLIC "-//Sun Microsystems, Inc.//DTD JSP Tag Library 1.2//EN"

"http://java.sun.com/j2ee/dtd/web-jsptaglibrary\_1\_2.dtd">

<taglib>

<tlib-version>1.0</tlib-version>

<jsp-version>1.2</jsp-version>

<short-name>MyTag</short-name>

<uri>my</uri>

<tag>

<name>choco</name>

<tag-class>my.first</tag-class>

<body-content>empty</body-content>

<attribute>

<name>texture</name>

<type>java.lang.String</type>

<required>true</required>

<rtexprvalue>true</rtexprvalue>

</attribute>

</tag>

</taglib>

Web.xml

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

<web-app version="2.5"

xmlns="http://java.sun.com/xml/ns/javaee"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://java.sun.com/xml/ns/javaee

http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd">

<jsp-config>

<taglib>

<taglib-uri>my</taglib-uri>

<taglib-location>/WEB-INF/tlds/mydate.tld</taglib-location>

</taglib>

</jsp-config>

</web-app>

Q7. . Create a custom tag “substring” with 3 mandatory attributes “input”, “start”, “end” which will do substring operation on given input

.jsp

<%@ taglib uri="my" prefix="MyTag" %>

<html>

<body>

Substring:&nbsp

<MyTag:substring input="Urvashi" start="1" end="5" />

</body>

</html>

.java

package my;

import javax.servlet.\*;

import javax.servlet.http.\*;

import javax.servlet.jsp.\*;

import javax.servlet.jsp.tagext.TagSupport;

import java.io.\*;

import java.util.\*;

public class first extends TagSupport

{ JspWriter out;

private String input;

private String start;

private String end;

int a,b;

public void setInput(String input)

{ this.input=input;

}

public void setStart(String start)

{ this.start=start;

a=Integer.parseInt(start);

}

public void setEnd(String end)

{ this.end=end;

b=Integer.parseInt(end);

}

public int doStartTag()throws JspException

{ out=pageContext.getOut();

return SKIP\_BODY;

}

public int doEndTag()throws JspException

{

try{ out.println(input.substring(a,b));

}

catch(Exception e)

{

try{

out.println(e);

}

catch(Exception ec)

{ ec.printStackTrace();

}

}

return EVAL\_PAGE;

}

}

.tld

<?xml version="1.0" encoding="ISO-8859-1" ?>

<!DOCTYPE taglib

PUBLIC "-//Sun Microsystems, Inc.//DTD JSP Tag Library 1.2//EN"

"http://java.sun.com/j2ee/dtd/web-jsptaglibrary\_1\_2.dtd">

<taglib>

<tlib-version>1.0</tlib-version>

<jsp-version>1.2</jsp-version>

<short-name>MyTag</short-name>

<uri>my</uri>

<tag>

<name>substring</name>

<tag-class>my.first</tag-class>

<body-content>empty</body-content>

<attribute>

<name>input</name>

<type>java.lang.String</type>

<required>true</required>

<rtexprvalue>true</rtexprvalue>

</attribute>

<attribute>

<name>start</name>

<type>java.lang.String</type>

<required>true</required>

<rtexprvalue>true</rtexprvalue>

</attribute>

<attribute>

<name>end</name>

<type>java.lang.String</type>

<required>true</required>

<rtexprvalue>true</rtexprvalue>

</attribute>

</tag>

</taglib>

Web.xml

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

<web-app version="2.5"

xmlns="http://java.sun.com/xml/ns/javaee"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd">

<jsp-config>

<taglib>

<taglib-uri>my</taglib-uri>

<taglib-location>/WEB-INF/tlds/mydate.tld</taglib-location>

</taglib>

</jsp-config>

</web-app>

Q8. Create a custom tag “reverse” with a mandatory attribute “input” to reverse a string.

.jsp

<%@ taglib uri="my" prefix="MyTag" %>

<html>

<body>

Reverse is :&nbsp

<MyTag:reverse input="URVASHI" />

</body>

</html>

.java

package my;

import javax.servlet.\*;

import javax.servlet.http.\*;

import javax.servlet.jsp.\*;

import javax.servlet.jsp.tagext.TagSupport;

import java.io.\*;

import java.util.\*;

public class first extends TagSupport

{ JspWriter out;

private String input;

public void setInput(String input)

{ this.input=input;

}

public int doStartTag()throws JspException

{ out=pageContext.getOut();

return SKIP\_BODY;

}

public int doEndTag()throws JspException

{

try{ char[] c=input.toCharArray();

ArrayList al=new ArrayList<Character>();

for(char e:c)

al.add(e);

Collections.reverse(al);

Iterator i=al.listIterator();

while(i.hasNext())

out.print(i.next());

}

catch(Exception e)

{

try{

out.println(e);

}

catch(Exception ec)

{ ec.printStackTrace();

}

}

return EVAL\_PAGE;

}

}

.tld

<?xml version="1.0" encoding="ISO-8859-1" ?>

<!DOCTYPE taglib

PUBLIC "-//Sun Microsystems, Inc.//DTD JSP Tag Library 1.2//EN"

"http://java.sun.com/j2ee/dtd/web-jsptaglibrary\_1\_2.dtd">

<taglib>

<tlib-version>1.0</tlib-version>

<jsp-version>1.2</jsp-version>

<short-name>MyTag</short-name>

<uri>my</uri>

<tag>

<name>reverse</name>

<tag-class>my.first</tag-class>

<body-content>empty</body-content>

<attribute>

<name>input</name>

<type>java.lang.String</type>

<required>true</required>

<rtexprvalue>true</rtexprvalue>

</attribute>

</tag>

</taglib>

Q9. Create a custom tag "today" that displays today's date and time

.jsp

<%@ taglib uri="my" prefix="MyTag" %>

<html>

<body>

<MyTag:today />

</body>

</html>

.java

package my;

import javax.servlet.\*;

import javax.servlet.http.\*;

import javax.servlet.jsp.\*;

import javax.servlet.jsp.tagext.TagSupport;

import java.io.\*;

import java.util.\*;

public class MyDate1 extends TagSupport

{ JspWriter out;

public int doStartTag()throws JspException

{ out=pageContext.getOut();

try{ //out.println("Start Tag");

}

catch(Exception e)

{

try{

out.println(e);

}

catch(Exception ec)

{ ec.printStackTrace();

}

}

return EVAL\_BODY\_INCLUDE;

}

public int doEndTag()throws JspException

{ String s=new Date().toString();

try{ out.println("Today's date and time: "+s);

}

catch(Exception e)

{

try{

out.println(e);

}

catch(Exception ec)

{ ec.printStackTrace();

}

}

return EVAL\_PAGE;

}

}

.tld

<?xml version="1.0" encoding="ISO-8859-1" ?>

<!DOCTYPE taglib

PUBLIC "-//Sun Microsystems, Inc.//DTD JSP Tag Library 1.2//EN"

"http://java.sun.com/j2ee/dtd/web-jsptaglibrary\_1\_2.dtd">

<taglib>

<tlib-version>1.0</tlib-version>

<jsp-version>1.2</jsp-version>

<short-name>MyTag</short-name>

<uri>my</uri>

<tag>

<name>today</name>

<tag-class>my.MyDate1</tag-class>

<body-content>JSP</body-content>

</tag>

</taglib>

Q10. Ask a user's name and age on a HTML form. Then display Hello <uname> on a JSP. On the same page ask the product the user would like to buy. Then redirect to another JSP which would display: Hello <uname>, You have ordered <product>. (Use Session Scope Variable using setTag)

.html

<html>

<body>

<form action="num.jsp">

Enter name<br>

<input type="text" name="t1"><br>

Enter age<br>

<input type="text" name="t2"><br>

<input type="submit">

</form>

</body>

</html>

.jsp

<%@ page contentType="text/html" %>

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

<html>

<body>

Hello&nbsp

<c:out value="${param.t1}"/>

<input type="hidden" name="submitted" value="true">

<c:if test="${param.submitted && empty param.userName}">

<tr><td></td>

<td colspan="2"><font color="red">

Please enter your Name

</font></td></tr>

</c:if>

Enter the product you would like to buy

<input type="text" name="product" value="<c:out value="${param.product}" />">

</body>

</html>

**JDBC**

Q1. Create Student and Results Database and perform the following using JDBC programs

a. Find total number of students

b. Print average marks for each subject input by user.

c. Find the name of student getting highest marks.

d. Find no of students getting first, second and third division.

e. Find subject wise toppers

f. Find the average marks

g. Find the student getting second highest marks.

import java.sql.\*;

import java.util.\*;

public class menu

{ public static void main(String args[])

{ try{ int i=0;

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

Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/student?autoReconnect=true&useSSL=false","root","urvashi@");

int ch;

while(true)

{

System.out.println("\n MENU:-\n 1 Find total number of students");

System.out.println("2 Print average marks for each subject input by user");

System.out.println("3 Find the name of student getting highest marks");

System.out.println("4 Find no of students getting first, second and third division");

System.out.println("5 Find subject wise toppers");

System.out.println("6 Find the average marks");

System.out.println("7 Find the student getting second highest marks");

System.out.println("8 Exit");

System.out.println("chose any one option");

Scanner s=new Scanner(System.in);

ch=s.nextInt();

PreparedStatement ps=null;

ResultSet rs=null;

switch(ch)

{ case 1:

ps=con.prepareStatement("select count(\*) as total\_no\_of\_Students from stu");

rs=ps.executeQuery();

while(rs.next())

{ i=rs.getInt("total\_no\_of\_Students");

System.out.println("Total no. of Students:"+i);

}

rs.close();

break;

case 2:

ps=con.prepareStatement("select TOC from stu");

rs=ps.executeQuery();

float sum=0;

while(rs.next())

{ sum+=rs.getInt("TOC");

}

System.out.println("Average marks for TOC="+sum/i);

rs.close();

ps=con.prepareStatement("select SP from stu");

rs=ps.executeQuery();

sum=0;

while(rs.next())

{ sum+=rs.getInt("SP");

}

System.out.println("Average marks for SP="+sum/i);

rs.close();

ps=con.prepareStatement("select IT from stu");

rs=ps.executeQuery();

sum=0;

while(rs.next())

{ sum+=rs.getInt("IT");

}

System.out.println("Average marks for IT="+sum/i);

rs.close();

ps=con.prepareStatement("select ALGO from stu");

rs=ps.executeQuery();

sum=0;

while(rs.next())

{ sum+=rs.getInt("ALGO");

}

System.out.println("Average marks for ALGO="+sum/i);

break;

case 3:

ps=con.prepareStatement("select name,MAX(total\_marks) from stu,result where stu.rollno=result.rollno");

rs=ps.executeQuery();

while(rs.next())

{ String n=rs.getString("name");

System.out.println("Student having highest marks:- "+n);

}

rs.close();

break;

case 4:

ps=con.prepareStatement("select division,count(\*) as no from result GROUP BY division");

rs=ps.executeQuery();

ResultSetMetaData rsm=rs.getMetaData();

int column=rsm.getColumnCount();

for(i=1;i<=column;i++)

System.out.print(rsm.getColumnName(i)+" ");

System.out.println();

while(rs.next())

{ for(i=1;i<=column;i++)

{ System.out.print(rs.getString(i)+" ");

}

System.out.println();

}

rs.close();

break;

case 5:

String n=null;

ps=con.prepareStatement("select name,MAX(TOC) from stu");

rs=ps.executeQuery();

while(rs.next())

{ n=rs.getString("name");

System.out.println("Topper in TOC:- "+n);

}

rs.close();

ps=con.prepareStatement("select name,MAX(SP) from stu");

rs=ps.executeQuery();

while(rs.next())

{ n=rs.getString("name");

System.out.println("Topper in SP:- "+n);

}

rs.close();

ps=con.prepareStatement("select name,MAX(IT) from stu");

rs=ps.executeQuery();

while(rs.next())

{ n=rs.getString("name");

System.out.println("Topper in IT:- "+n);

}

rs.close();

ps=con.prepareStatement("select name,MAX(ALGO) from stu");

rs=ps.executeQuery();

while(rs.next())

{ n=rs.getString("name");

System.out.println("Topper in ALGO:- "+n);

}

rs.close();

break;

case 6:

int n1=0;

ps=con.prepareStatement("select count(\*) as no from result");

rs=ps.executeQuery();

while(rs.next())

{ n1=rs.getInt("no");

}

rs.close();

ps=con.prepareStatement("select SUM(total\_marks) as t from result");

rs=ps.executeQuery();

while(rs.next())

{ int in=rs.getInt("t");

System.out.println("Average Marks:- "+(float)in/n1);

}

rs.close();

break;

case 7:

ps=con.prepareStatement("select name from stu,result where stu.rollno=result.rollno order by total\_marks desc limit 1,1");

rs=ps.executeQuery();

while(rs.next())

{

String n2=rs.getString("name");

System.out.println("Student having second highest marks:- "+n2);

}

rs.close();

break;

default:

con.close();

System.exit(0);

}

}

}

catch(Exception e)

{ System.out.println("EXCEPTION :"+e);

}

}

}

Q2. Create a procedure in MySQL to count the number of Rows in table 'Student'. Use CallableStatement to call this method from Java code.

import java.sql.\*;

public class callable

{ public static void main(String args[])

{ try{ String i=null;

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

System.out.println("Driver Registered");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/student?autoReconnect=true&useSSL=false","root","urvashi@");

String query="call countstudent(?)";

CallableStatement cs=con.prepareCall(query);

cs.registerOutParameter(1,Types.INTEGER);

cs.execute();

System.out.println("Total no. of Students = "+cs.getInt(1));

con.close();

}

catch(Exception e)

{ System.out.println("EXCEPTION :"+e);

}

}

}

**JAVA BEANS**

Q1. Implement Student JavaBean using Serializability Interface.

package mypack;

import java.io.\*;

public class Employee\_bean implements Serializable

{ private int id;

private String name;

public Employee\_bean()

{

}

public void setId(int id)

{ this.id=id;

}

public void setName(String name)

{ this.name=name;

}

public int getId()

{ return id;

}

public String getName()

{ return name;

}

}

Q2. Implement Employee JavaBean using Serializability Interface.

package mypack;

import java.io.\*;

public class Student\_bean implements Serializable

{ int rollno;

int[] marks;

String name;

public void setRollno(int r)

{ this.rollno=r;

}

public void setName(String name)

{ this.name=name;

}

public void setMarks(int m,int index)

{ marks[index]=m;

}

public int getRollno()

{ return rollno;

}

public String getName()

{ return name;

}

int getMarks(int index)

{ return marks[index];

}

void setMarks(int[] m)

{ marks=m;

}

}

**JAVA**

Q1. Implement a Bank Account having Instance variables: Account Number, Balance and

having methods:

float Deposit (float x)

float withdraw (float x)

int get account no ()

float get balance ()

tax deduction ()

Then implement class Bank having an array list of accounts of type BankAccount.

Implement following methods:

AddAccount in Bank

Get Total balance in Bank

Get account number with max. and min. balance

Find an account given a bank account no.

Count no. of accounts having atleast specific balance

class BankAccount

{long accno;

long bal;

static count=56789231;

BankAccount()

{ this.bal=0;

this.accno=count;

count++;

}

float deposit(float x)

{ this.bal+=x;

}

float withdraw(float x)

{ this.bal-=x;

}

float

int get\_account\_no()

{ return accno;

}

float get\_balance()

{ return bal;

}

float tax\_deduction(float rate)

{ float r=rate/100;

return r\*bal;

}

}

class Bank

{ ArrayList<BankAccount> al;

Bank()

{ al=new ArrayList<BankAccount>;

}

void add\_acc(BankAccount ac)

{ al.add(ac);

}

float get\_total\_bal()

{ float total;

for(BalanceAccount x:al)

{ total+=x.get\_balance();

}

}

long get\_max()

{ float max=al.get\_balance();

for(BalanceAccount x:al)

{ if(x.get\_balance>max)

{ max=x.get\_balance();

}

}

return x.accno;

}

long get\_min()

{ float min=al.get\_balance();

for(BalanceAccount x:al)

{ if(x.get\_balance<min)

{ min=x.get\_balance();

}

}

return x.accno;

}

BankAccount find\_account(int accno)

{ for(BalanceAccount x:al)

{ if(x.get\_account\_no=accno)

{ return x;

}

}

return null;

}

int countAccount(float balance)

{ int count=0;

for(al.get\_balance()>=balance)

{ count++;

}

return count;

}

}

class ex

{ public static void main(String args[])

{ Scanner s=new Scanner(System.in);

System.out.println("Enter your choice");

System.out.println("1.Add an account \n 2.Deposit \n3.Withdraw ");

int a=s.nextInt();

}

}

Q2. Implement an Abstract Class Stack with methods push, pop, display for two classes: StaticStack and DyanamicStack. StaticStack uses one dimensional integer array to store numbers and DyanamicStack uses an integer ArrayList to store.

import java.util.\*;

abstract class Stack

{

int top;

Stack()

{ top=-1;

}

abstract void push(int ele);

abstract int pop();

abstract void display();

}

class StaticStack extends Stack

{ int[] a;

StaticStack(int size)

{ a=new int[size];

}

void push(int ele)

{ a[++top]=ele;

}

int pop()

{ return a[top--];

}

void display()

{ for(int i=0;i<=top;i++)

{ System.out.println(a[i]+"\n");

}

}

}

class DynamicStack extends Stack

{ ArrayList<Integer> al;

DynamicStack()

{ al=new ArrayList<Integer>();

}

void push(int ele)

{ ++top;

al.add(ele);

}

int pop()

{

al.remove(top--);

return 0;

}

void display()

{ System.out.println("Contents of stack are:"+al);

}

}

class ex

{ public static void main(String[] args)

{ System.out.println("enter your choice\n 1.STATIC STACK \n 2.DYNAMIC STACK");

int a;

Scanner sc=new Scanner(System.in);

a=sc.nextInt();

switch(a)

{

case 1:

StaticStack s=new StaticStack(5);

System.out.println(" STATIC STACK ");

for(int i=0;i<5;i++)

{ System.out.println("(push) enter any no.");

s.push(sc.nextInt());

}

s.display();

s.pop();

System.out.println("after pop");

s.display();

break;

case 2:

System.out.println(" DYNAMIC STACK ");

DynamicStack ds=new DynamicStack();

for(int i=0;i<5;i++)

{ System.out.println("(push) enter any no.");

ds.push(sc.nextInt());

}

ds.display();

ds.pop();

System.out.println("after pop");

ds.display();

break;

}

}

}

**JAVASCRIPT**

Q1. . Create a student registration form. Create functions to perform the following checks:

a. Roll number is a 7 digit numeric value

b. Name should be an alphabetical value (String)

c. DOB entered in dd/mm/yy format and should be display in words (e.g. Saturday, January 01, 2000)

d. Check on non-empty fields

<html>

<head>

<script language="Javascript">

function signup()

{

var vname=document.frmsignup.txtname.value;

var vrollno=document.frmsignup.txtrno.value;

if((vname!="") && (vrollno!=""))

{

<!-- if(isString(vname)) -->

alert("Name is "+vname);

<!-- else -->

<!-- alert("name should not contain number"); -->

if(vrollno.length==7)

{

if(isNaN(vrollno)==false)

{

alert("Roll number is "+vrollno);

}

else

alert("ROLL NUMBER SHOULD CONTAIN ONLY DIGITS......RE-ENTER");

}

else

alert("Roll number should be seven digit......RE-ENTER");

var vdob;

var vmonth=document.frmsignup.slcmonth.value;

switch(vmonth)

{

case "1":month="January";

break;

case "2":month="February";

break;

case "3":month="March";

break;

case "4":month="April";

break;

case "5":month="May";

break;

case "6":month="June";

break;

case "7":month="July";

break;

case "8":month="August";

break;

case "9":month="September";

break;

case "10":month="October";

break;

case "11":month="November";

break;

case "12":month="December";

break;

}

vdob=month+" "+document.frmsignup.slcdate.value+","+document.frmsignup.slcyear.value;

alert("Dob is "+vdob);

}

else

{

alert("FILL ALL THE TEXT FIELDS...");

}

}

</script>

</head>

<body>

<form name="frmsignup" method="post">

NAME:

<input type="text" name="txtname"/>

<br><br>

ROLL NO:

<input type="text" name="txtrno"/>

<br><br>

Date of Birth:

<br>

Date:<select name="slcdate">

<script language="Javascript">

for(var i=1;i<=31;i++)

{

document.write("<option value="+i+">");

document.write(i);

document.write("</option>");

}

</script>

</select>

Month:<select name="slcmonth">

<script language="Javascript">

for(var i=1;i<=12;i++)

{

document.write("<option value="+i+">");

document.write(i);

document.write("</option>");

}

</script>

</select>

Year:<select name="slcyear">

<script language="Javascript">

for(var i=1917;i<=2017;i++)

{

document.write("<option value="+i+">");

document.write(i);

document.write("</option>");

}

</script>

</select>

<br><br><br>

<input type="submit" onClick = "signup()"/>

</form>

</body>

</html>

Q2. Implement a Static Password Protection.

<html>

<head>

<script language="javascript">

function check()

{

var user=new Array("abc","pqr","xyz");

var pswd=new Array("123","234","456");

var u=document.form1.text1.value;

var p=document.form1.text2.value;

var flag=0;

for(var i=0;i<user.length;i++)

{

if(u==user[i])

if(p==pswd[i])

{

alert("Valid details!!! WELCOME");

flag=1;

break;

}

}

if(flag==0)

alert("TRY AGAIN!!!");

}

</script>

</head>

<body>

<form name="form1" method="post">

Enter Username :

<input type="text" name="text1"/> <br><br>

Enter Password :

<input type="text" name="text2"/> <br><br><br>

<input type="submit" value="CHECK" onClick="check()"/>

</form>

</body>

</html>

Q3. Write a java script to sort an array using bubble sort. Take the number of elements and array from user.

<html>

<head><script>

var size=parseInt(prompt("Enter the size of array "));

var arr=new Array();

for(var i=0;i<size;i++)

arr[i]=parseInt(prompt("enter "+i+"th element"));

document.write(" Size of entered array is : "+ size +" <br><br>");

document.write(" Entered array is : <br>");

for(var i=0;i<size;i++)

document.write(arr[i]+" ");

for(var i=0;i<size;i++)

{

for(var j=0;j<size-i-1;j++)

if(arr[j]>arr[j+1])

{

var temp=arr[j];

arr[j]=arr[j+1];

arr[j+1]=temp;

}

}

document.write("<br> <br>");

document.write(" Sorted array is : <br>");

for(var i=0;i<size;i++)

document.write(arr[i]+" ");

</script>

</head>

</html>

Q4. Write a JavaScript to implement stack methods (push and pop).

<html>

<head>

<script language="Javascript">

function push(elm)

{

this.top++;

this.data[this.top]=elm;

}

function display()

{

for(var i=0;i<=this.top;i++)

document.write(this.data[i]+" ");

document.write("<br>");

}

function pop()

{

if(this.top>=0)

return this.data[this.top--];

else

return null;

}

function stack()

{

this.top=-1;

this.data=new Array();

this.push=push;

this.display=display;

this.pop=pop;

}

var s1=new stack();

var ans='y';

var n= parseInt(prompt("how many elements u want to push..."));

var stack=new Array();

for(var i=0;i<n;i++)

stack[i]=parseInt(prompt("enter "+i+"th element to push"));

for (var i=0;i<n;i++)

s1.push(stack[i]);

s1.display();

s1.pop();

s1.display();

</script>

</head>

<body>

</body>

</html>

Q5. Write a JavaScript

1. to change the color of text using setTimeOut()

<html>

<head>

<script language="javascript">

function setToRed ( )

{

document.getElementById("txt").style.color = "#FF0000";

setTimeout ( "setToBlack()", 2000 );

}

function setToBlack ( )

{

document.getElementById("txt").style.color = "#000000";

}

</script>

</head>

<body>

Enter Text:

<form name="form1" method="post">

<textarea id="txt" rows="5" cols="50"> </textarea> <br>

<br><br>

<input type="button" value="Click me and see!" onclick="setToRed()"/>

</form>

</body>

</html>

1. to move an image across screen using setInterval()

<html>

<head>

<title>JavaScript Animation</title>

<script type="text/javascript">

function init(){

imgObj = document.getElementById('myImage');

imgObj.style.position= 'relative';

imgObj.style.left = '0px';

setInterval( "moveRight()", 1000 );

}

function moveRight()

{

imgObj.style.left = parseInt(imgObj.style.left) + 50 + 'px';

}

</script>

</head>

<body>

<form>

<img id="myImage" src="C:\Users\Public\Pictures\Sample Pictures\Tulips.jpg" width="100" height="100"/>

<p>Click button below to move the image to right</p>

<input type="button" value="Click Me" onclick="init()" />

</form>

</body>

</html>

Q6. Implement the question no. 1 of hands on exercises of chapter 10 (page 190).

<html>

<head>

<title> Elements of a form </title>

<script language="javascript">

function UserInfo()

{

var fname=document.info\_form.fname.value;

var lname=document.info\_form.lname.value;

var Email=document.info\_form.email.value;

var Address=document.info\_form.address.value;

var City=document.info\_form.city.value;

var State=document.info\_form.state.value;

var P\_code=document.info\_form.p\_code.value;

var Country=document.info\_form.country.value;

if((fname!="") && (lname!="") && (Email!="") && (Address!="") && (City!="") && (State!="") && (P\_code!="") && (Country!=""))

{

alert("User Information is : \n First Name : "+fname+"\n Last Name : "+lname+"\n E-mail Address : "+Email+"\n Address : "+Address+"\n City : "+City+"\n State : "+State+"\n Postal Code : "+P\_code+"\n Country : "+Country);

<!-- Getting value of selected radio button -->

var radios = document.getElementsByName('statement');

for (var i=0, len=radios.length; i<len; i++)

if ( radios[i].checked )

{

alert("The statement you chose : "+radios[i].value);

break;

}

<!-- Getting the value of selected check boxes -->

var cbox = document.getElementsByName('slcbox');

for (var i=0, len=cbox.length; i<len; i++)

if ( cbox[i].checked )

alert("Your interests are : "+cbox[i].value);

<!-- Getting the value of selected item from drop down list -->

var e = document.getElementById("slc");

var strUser = e.options[e.selectedIndex].text;

alert("You have learned about our website from : "+strUser);

<!-- Getting the user comments -->

var Comments=document.info\_form.comments.value;

if(Comments!=" ")

alert("Your comments about us are : "+Comments);

}

else

{

alert("Please fill out all the text-boxes with entries... \n Text-Boxes entry is necessary for your entry to be processed...");

}

}

</script>

</head>

<body>

<B> <I><h1> INFONET SERVICES </h1></I>

<form name="info\_form" method="POST">

<input type="text" name="fname"/> First Name <br><br>

<input type="text" name="lname"/> Last Name <br><br>

<input type="text" name="email"/> E-mail Address <br><br>

<input type="text" name="address"/> Address <br><br>

<input type="text" name="city"/> City <br><br>

<input type="text" name="state"/> State

<input type="text" name="p\_code"/> Postal Code

<input type="text" name="country"/> Country <br><br>

<I> Please choose the most appropriate statement </I> <br>

<input type="radio" name="statement" value="I regularly purchase items online"/> I regularly purchase items online <br>

<input type="radio" name="statement" value="I have on occasion purchased items online"/> I have on occasion purchased items online <br>

<input type="radio" name="statement" value="I have not purchased anything online, but I would consider it"/> I have not purchased anything online, but I would consider it <br>

<input type="radio" name="statement" value="I prefer to shop in real stores"/> I prefer to shop in real stores <br><br><br>

<I>I'm interested in (choose all that apply) <br> </I>

<input type="checkbox" name="slcbox" value="Hiking"/> Hiking <br>

<input type="checkbox" name="slcbox" value="Mountain Biking"/> Mountain Biking <br>

<input type="checkbox" name="slcbox" value="Camping"/> Camping <br>

<input type="checkbox" name="slcbox" value="Rock Climbing"/> Rock Climbing <br>

<input type="checkbox" name="slcbox" value="Off-Road 4WD"/> Off-Road 4WD <br>

<input type="checkbox" name="slcbox" value="Cross-country Skiing"/> Cross-country Skiing <br>

<br><br>

<I> I learned about this site from </I> <br>

<select id="slc">

<option value="Print Ads"/> Print Ads

<option value="Online"/> Online

<option value="From Friends"/> From Friends

</select>

<br><br><br>

Comments <br>

<textarea name="comments" rows="5" cols="50"> </textarea> <br>

<br><br>

<input type="submit" value="Submit" onClick="UserInfo()"/>

<input type="submit" value="Start Over"/> </B>

</form>

</body>

</html>