**WEB TECHNOLOGY**

**LAB MANUAL (KCS-652)**

**Session:-**2021-2022

**B.TECH THIRD YEAR**



**Department of Computer Science and Engineering, Bundelkhand Institute of Engineering and Technology, Jhansi (U.P.) India – 284128**

## Submitted To: Submitted By:

## Er. Umakant Ahirwar Vidhi Mishra(1904310063)

## 

**INDEX**

**Name:** Vidhi Mishra **Roll No. :** 1904310063

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No.** | **Date** | **Aim/Objective** | **T. Sign &**  **Remarks** |
| 1 | 02/02/2022 | Write HTML/Java scripts to display your CV in navigator, your Institute website, Department Website and Tutorial website for specific subject |  |
| 2 | 09/02/2022 | Write an HTML program to design an entry form of student details and send it to a database server like SQL,Oracle or MS Access. |  |
| 3 | 23/02/2020 | Write programs using Javascript for Web Page to display browsers information. |  |
| 4 | 02/03/2022 | Write a Java applet to display the Application Program screen i.e. calculator. |  |
| 5 | 09/03/2022 | Write a program in XML for creation of DTD, which specifies set of rules. Create a stylesheet in CSS/ XSL & display the document in internet explorer. |  |
| 6 | 16/03/2022 | Program to illustrate JDBC connectivity. Program for maintaining databases by sending queries. Design and implement a simple servlet book query with the help of JDBC & SQL. Create MS Access Database, create on ODBC link, Compile & execute JAVA JDVC Socket. |  |
| 7 | 23/03/2022 | Install TOMCAT web server and APACHE. Access the above developed static web pages for books website, using these servers by putting the web pages developed. |  |
| 8 | 30/03/2022 | Assume four users user1, user2, user3 and user4 having the passwords pwd1, pwd2, pwd3 and pwd4 respectively. Write a servlet for doing the following.1. Create a Cookie and add these four user id’s and passwords to this Cookie. Login form and authenticate with the values available in the cookies. |  |
| 9 | 06/04/2022 | Install a database (Mysql or Oracle). Create a table which should contain at least the following fields: name, password, email-id, phone number Write a java program/servlet/JSP to connect to that database and extract data from the tables and display them. Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page. |  |
| 10 | 13/04/2022 | Write a JSP which inserts the details of the 3 or 4 users who register with the web site by using registration form. Authenticate the user when he submits the login form using the username and password from the database. |  |

**Practical 1**

**Write HTML/Java scripts to display your CV in navigator, your Institute website, Department Website and Tutorial website for specific subject**

**Code of the HTML file:**

<!DOCTYPE html>

<html>

<head>

<meta name="viewport"

content="width=device-width, initial-scale=1">

<title>Home page</title>

<link rel="stylesheet" href="all.css">

</head>

<body style="background-color: #f9f7f7;" leftmargin="50">

<div class="header">

<table align="center">

<tr>

<td>

<a href="http://bietjhs.ac.in"><img src="bietlogo.png" width="150" align="left"></a>

</td>

<td>

<a href="http://bietjhs.ac.in"><p class="x">Bundelkhand Institute Of Engineering And Technology, Jhansi</p></a>

</td>

<td>

<a href="http://bietjhs.ac.in"><img src="b1.jpg" width="200" align="right"></a>

</td>

</tr>

</table>

</div>

<ul class="sticky">

<li class="m"><a

href="home.html">Home</a></li>

<li class="m"><a href="clg.html">My College</a></li>

<li class="m"> <a href="faculty.html">My Faculties</a></li>

<li class="m"> <a href="scoit.html">SCOIT</a></li>

</ul> <BR>

<div style="padding:70px 200px; height:10px;float: right;">

<img class="r" src="ad.png" style="width:200px" height="160px" alt="Your DP">

</div>

<div style="margin-left:5%;padding:1px 16px;height:1000px;margin-right: 5%;"> <br>

<p class="h">Abc</p>

<p class="B">I am a BTech third Year student studying with branch

<br>INFORMATION TECHNOLOGY at

Bundelkhand Institute <br>Of Engineering And Technology, Jhansi</p>

<br><br>

<p class="h">MY PERSONAL INFORMATION<hr></p>

<ul type="disc" class="p">

<li><p class="B">xyz</p></li>

<li><p class="B">I am studying from the

college named Bundelkhand Institute Of Engineering And Technology, situated in Jhansi.</p></li>

<li><p class="B">I am in my third year with branch INformation Technology.</p></li>

<li><p class="B">My date of Birth is November 0000. And hence, my age is 000.</p></li>

<li><p class="B">I know english and hindi fluently.</p></li>

<li><p class="B">Technically, I know coding in various languages like C++, C, Java, python, html and css.</p></li>

<li><p class="B">My daily hobbies are Singing, sketching and reading books, </p></li>

</ul><br>

<br><p class="h">Previous Examinations<hr></p>

<ul type="disc" class="p">

<li><p class="B">I have completed my schooling from BBD Sr.School UP.</p></li>

<li><p class="B">My school was affiliated to Central Board Of Secondaty Education (CBSE).</p></li>

<li><p class="B">I passed my Tenth Grade in 000 scoring 00 CGPA. </p></li>

<li><p class="B">I passed my Twelth Grade in 000 with 000. </p></li>

<li><p class="B">I cleared the UPSee exam with a general rank of .</p></li>

<li><p class="B">I scored 000 YGPA in final exams of my first year.</p></li>

</ul><br>

<br><p class="h">Journey to BIET<hr></p>

<p class="Ba">I cleared the UPSEE exam with a general rank of . By this rank I got admission in my college for the degree of BTech. At present I am in third Year, and already like my studies here.<br>

I am here to build my career and seek for higher studies after completing my BTech here.

I scored 0000 YGPA in final exams of my second year. And aspires to score better and better in my future, in the college.

</p>

<br><br><br><br><br>

</div>

<div class="footer">

<p>Designed by xyz, Technologies used:HTML and CSS</p>

</div>

</body>

</html>

**Code For CSS:**

body { margin: 0;

font-family: Arial, Helvetica, sans-serif;

}

.header {

overflow: hidden;

margin:0;

background-color: #222831;

text-align: center;

padding: 1 px;

border-bottom:1px solid white;

}

.header a {

float: left;

color: white;

text- align: center;

text-decoration: none;

font- size: 30px;

line-height: 25px;

border-radius: 4px;

}

body

13

{

margin: 0;

}

@media screen and (max-width: 500px)

{

.header a

{

float: none;

display: block;

text-align: left;

}

}

p.B

{

color: #169ba5;

text-align: justify;

font- family: 'Cambria', serif;

font-size: 24px;

font-weight: normal;

line-height: 32px;

margin: 0 0 1px;

text- shadow: 1px 1px 1px #9bedf3;

}

p.Ba

{

color: #169ba5;

text-align: justify;

font-family: 'Cambria', serif;

font-size: 24px;

font-weight: normal;

line-height: 32px;

margin:1px 30px;

text-shadow: 1px 1px 1px #9bedf3;

}

p.x

{

14

font-family: 'Raleway',sans-serif;

font-size: 30px;

font-weight: bold;

margin- right: 60px;

color: whitesmoke;

text- shadow: 2px 4px 6px #111010;

text-align: center;

}

ul.p

{

list-style-type: disc;

color: #19adb8;

}

img.r

{

width: 210px;

height: 210px; border:2px solid #fff; box- shadow: 4px 6px 5px #ccc; border- radius:210px;

}

ul.sticky

{

list-style-type: none;

margin: 0;

padding:0;

overflow: hidden;

background-color: #222831;

border- bottom:1px solid white;

position: -webkit- sticky;

position: sticky;

top: 0;

}

li.m

{

float: left;

border-right:1px solid #bbb;

}

li.m:last-child

15

{

border-right: none;

}

li.m a

{

display: block;

color: white;

text-align: center;

padding: 14px 16px;

text- decoration: none;

text-shadow: 0px 1px 2px whitesmoke;

}

li.m a:hover

{

background-color: #555;

border-radius: 4px;

}

img.r

{

width: 210px;

height: 210px;

border:2px solid #fff;

box-shadow: 4px 6px 5px #ccc;

border-radius:210px;

}

.footer

{

font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans-serif;

position: fixed;

font-size: small;

left: 0;

bottom: 0;

width: 100%;

background-color: #222831;

color: white;

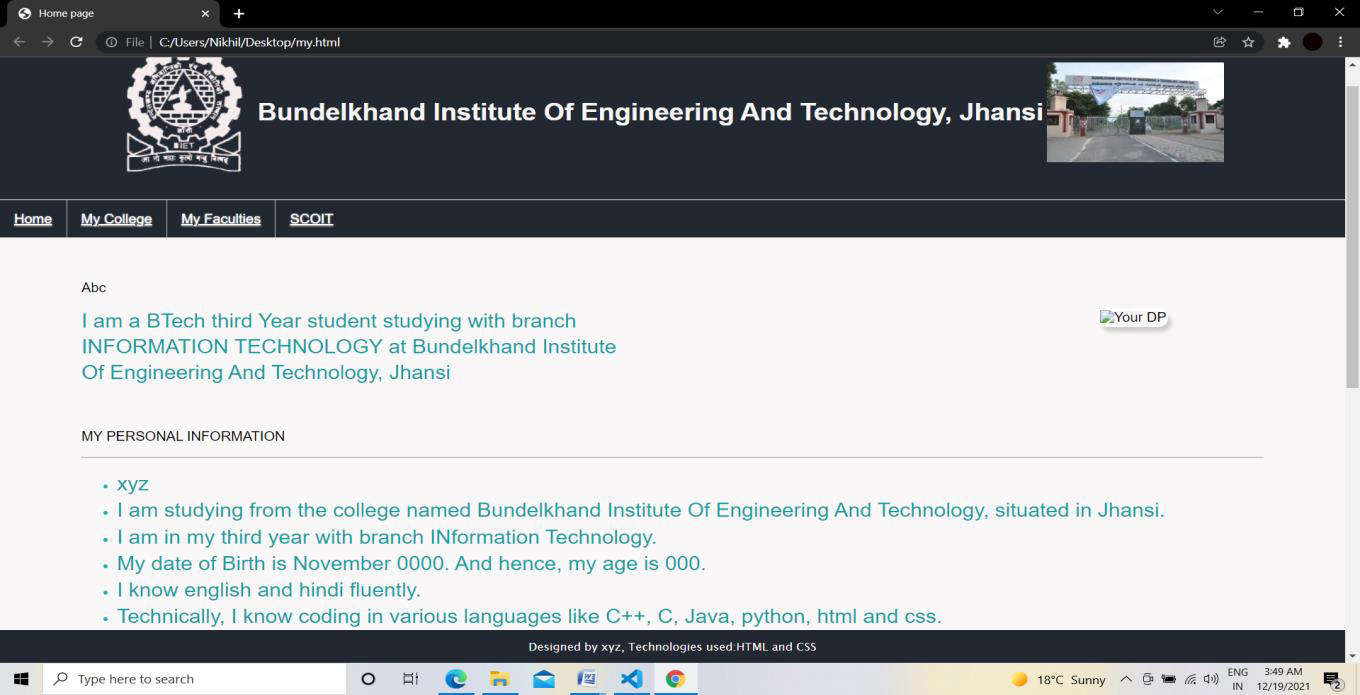
16

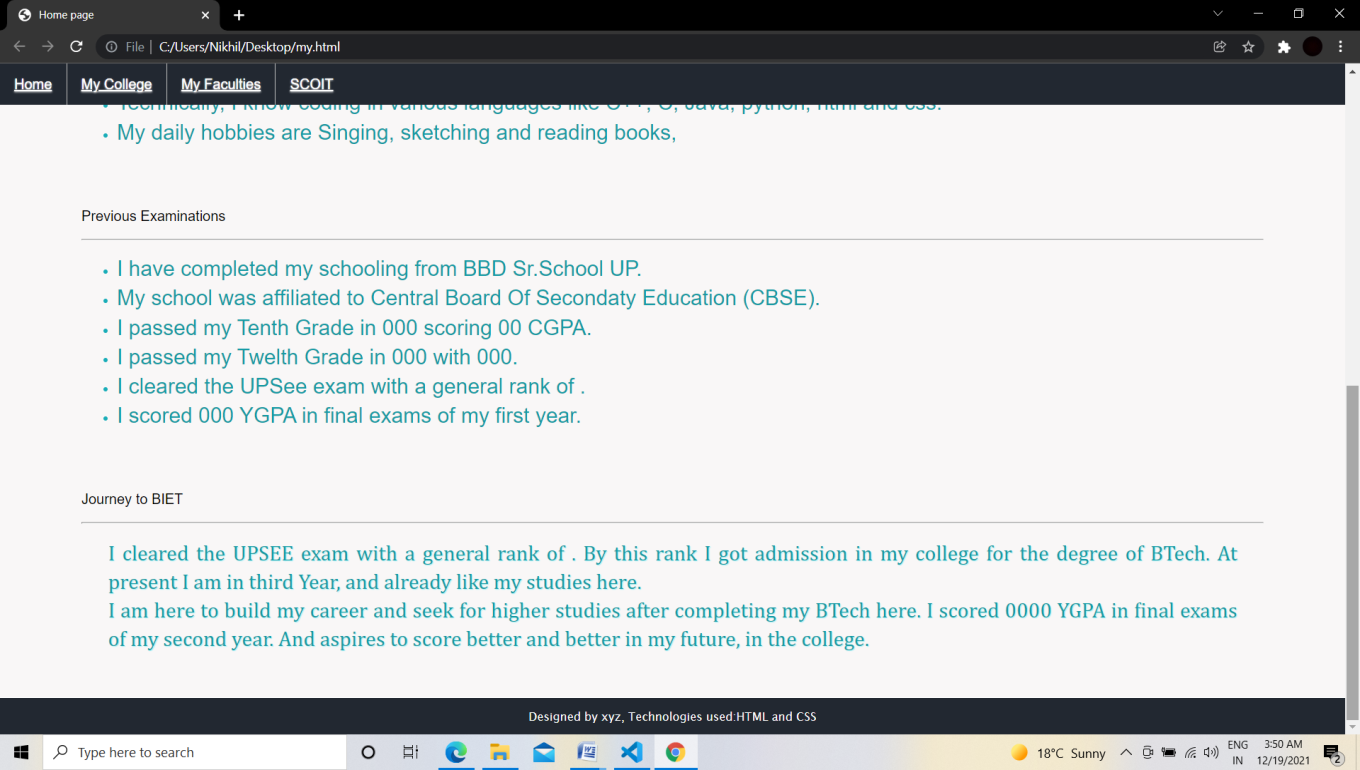
text-align: center;

border-top:1px solid white;

}

**Output:**

****



**Practical 2**

**Write an HTML program to design an entry form of student details and send it to a database server like SQL,Oracle or MS Access.**

<!DOCTYPE html>

<html>

<head>

<title>PHP insertion</title>

<link href="css/insert.css" rel="stylesheet">

</head>

<body>

<div class="maindiv">

<!--HTML Form -->

<div class="form\_div">

<div class="title">

<h2>Insert Data In Database Using PHP.</h2>

</div>

<form action="insert.php" method="post">

<!-- Method can be set as POST for hiding values in URL-->

<h2>Form</h2>

<label>Name:</label>

<input class="input" name="name" type="text" value="">

<label>Email:</label>

<input class="input" name="email" type="text" value="">

<label>Contact:</label>

<input class="input" name="contact" type="text" value="">

<label>Address:</label>

<textarea cols="25" name="address" rows="5"></textarea><br>

<input class="submit" name="submit" type="submit" value="Insert">

</form>

</div>

</div>

</body>

</html>

<?php

$connection = mysql\_connect("localhost", "root", ""); // Establishing Connection with Server

$db = mysql\_select\_db("colleges", $connection); // Selecting Database from Server

if(isset($\_POST['submit'])){ // Fetching variables of the form which travels in URL

$name = $\_POST['name'];

$email = $\_POST['email'];

$contact = $\_POST['contact'];

$address = $\_POST['address'];

if($name !=''||$email !=''){

//Insert Query of SQL

$query = mysql\_query("insert into students(student\_name, student\_email, student\_contact, student\_address) values ('$name', '$email', '$contact', '$address')");

echo "<br/><br/><span>Data Inserted successfully...!!</span>";

}

else{

echo "<p>Insertion Failed <br/> Some Fields are Blank....!!</p>";

}

}

mysql\_close($connection); // Closing Connection with Server

?>

**Practical 3**

**Write programs using Javascript for Web Page to display browsers information.**

**Code:**

<html>

<head>

<title>Browser Information</title>

</head>

<body bgcolor = “Cornsilk”>

<h1><u>Browser Information</u></h1>

<hr>

<ul>

<script LANGUAGE="JavaScript" type="text/javascript"> document.write("<li><b>Code Name:</b> " + navigator.appCodeName); document.write("<li><b>App Name:</b> " + navigator.appName); document.write("<li><b>App Version:</b> " + navigator.appVersion); document.write("<li><b>User Agent:</b> " + navigator.userAgent); document.write("<li><b>Language:</b> " + navigator.language); document.write("<li><b>Platform:</b> " + navigator.platform);

</script>

</ul>

<hr>

</body>

</html>

**Output:**

****

**Practical 4**

**Write a Java applet to display the Application Program screen i.e. calculator.**

**Code:**

import java.awt.\*;

import java.applet.\*;

import java.awt.event.\*;

public class Calculator extends Applet implements ActionListener

{

TextField inp;

public void init()

{

setBackground(Color.grey);

setLayout(null);

int i;

inp = new TextField();

inp.setBounds(150,100,270,50);

this.add(inp);

Button button[] = new Button[10];

for(i=0;i<10;i++)

{

button[i] = new Button(String.valueOf(9-i));

button[i].setBounds(150+((i%3)\*50),150+((i/3)\*50),50,50);

this.add(button[i]);

button[i].addActionListener(this);

}

Button dec=new Button(".");

dec.setBounds(200,300,50,50);

this.add(dec);

dec.addActionListener(this);

Button clr=new Button("C");

clr.setBounds(250,300,50,50);

19

this.add(clr);

clr.addActionListener(this);

Button operator[] = new Button[5];

operator[0]=new Button("/");

operator[1]=new Button("\*");

operator[2]=new Button("-");

operator[3]=new Button("+");

operator[4]=new Button("=");

for(i=0;i<4;i++)

{

operator[i].setBounds(300,150+(i\*50),50,50);

this.add(operator[i]);

operator[i].addActionListener(this);

}

operator[4].setBounds(350,300,70,50);

this.add(operator[4]);

operator[4].addActionListener(this);

}

String num1="";

String op="";

String num2="";

public void actionPerformed(ActionEvent e)

{

String button = e.getActionCommand();

char ch = button.charAt(0);

if(ch>='0' && ch<='9'|| ch=='.')

{

if (!op.equals(""))

num2 = num2 + button; else

num1 = num1 + button;

inp.setText(num1+op+num2);

}

20

else if(ch=='C')

{

num1 = op = num2 = "";

inp.setText("");

}

else if (ch =='=')

{

if(!num1.equals("") &&

!num2.equals(""))

{

double temp;

double n1=Double.parseDouble(num1);

double n2=Double.parseDouble(num2);

if(n2==0 && op.equals("/"))

{

inp.setText(num1+op+num2+" = Zero Division Error");

num1 = op = num2 = "";

}

if (op.equals("+"))

temp = n1 + n2;

else if (op.equals("-"))

temp = n1 - n2;

else if (op.equals("/"))

temp = n1/n2;

else

temp = n1\*n2;

inp.setText(num1+op+num2+" =

"+temp);

num1 = Double.toString(temp);

op = num2 = "";

} }

21

else

{

num1 = op = num2 = "";

inp.setText("");

}

}

{

if (op.equals("") || num2.equals(""))

op = button;

else

{

double temp;

double n1=Double.parseDouble(num1);

double n2=Double.parseDouble(num2);

if(n2==0 && op.equals("/"))

{

inp.setText(num1+op+num2+" = Zero Division Error");num1 = op = num2 = "";

}

else { if (op.equals("+"))

temp = n1 + n2;

else if (op.equals("-"))

temp = n1 - n2; else if (op.equals("/"))

temp = n1/n2; else

temp = n1\*n2;

num1 = Double.toString(temp);

op = button;

num2 = ""; }

}

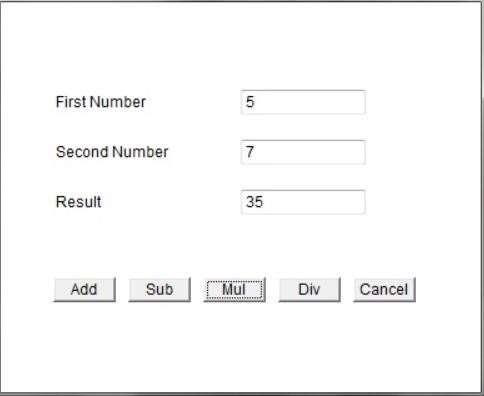
inp.setText(num1+op+num2);

}

}

}

**Output:**

****

**Practical 5**

**Write a program in XML for creation of DTD, which specifies set of rules. Create a stylesheet in CSS/ XSL & display the document in internet explorer.**

**Code:**

XML File

<?xml-stylesheet href="classic.css"?>

<ARTICLE>

<HEADLINE>Lorem Ipsum</HEADLINE>

<AUTHOR>Latin literature</AUTHOR>

<PARA>

Lorem Ipsum is simply dummy text of the printing and typesetting industry

<INSTRUMENT>lorem

ipsum</INSTRUMENT>

is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout.

</PARA>

</ARTICLE>

CSS File ARTICLE {

font-family: serif; background: khaki;

}

AUTHOR {

font-size: large; margin: 1em 0

}

HEADLINE {

font-size: x-large; margin-bottom: 1em

}

PARA

{

text-indent: 1em; text-align: justify

} INSTRUMENT

{

font-style: italic

}

**Practical 6**

**Program to illustrate JDBC connectivity. Program for maintaining databases by sending queries. Design and implement a simple servlet book query with the help of JDBC & SQL. Create MS Access Database, create on ODBC link, Compile & execute JAVA JDVC Socket.**

**Theory:**

**Creating JDBC Application:**

There are following six steps involved in building a JDBC application −

* **Import the packages:** Requires that you include the packages containing the JDBC classes needed for database programming. Most often, using import java.sql.\* will suffice.
* **Register the JDBC driver:** Requires that you initialise a driver so you can open a communication channel with the database.
* **Open a connection:** Requires using the DriverManager.getConnection() method to create a Connection object, which represents a physical connection with the database.
* **Execute a query**: Requires using an object of type Statement for building and submitting an SQL statement to the database.
* **Extract data from result set:** Requires that you use the appropriate ResultSet.getXXX() method to retrieve the data from the result set.
* **Clean up the environment:** Requires explicitly closing all database resources versus relying on the JVM's garbage collection.

STEPS:

//STEP 1. Import required packages

import java.sql.\*;

public class FirstExample {

// JDBC driver name and database URL

static final String JDBC\_DRIVER = "com.mysql.jdbc.Driver";

static final String DB\_URL = "jdbc:mysql://localhost/EMP";

// Database credentials

static final String USER = "username";

static final String PASS = "password";

public static void main(String[] args) {

Connection conn = null;

Statement stmt = null;

try{

//STEP 2: Register JDBC driver

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

//STEP 3: Open a connection

System.out.println("Connecting to database...");

conn = DriverManager.getConnection(DB\_URL,USER,PASS);

//STEP 4: Execute a query

System.out.println("Creating statement...");

stmt = conn.createStatement();

String sql;

sql = "SELECT id, first, last, age FROM Employees";

ResultSet rs = stmt.executeQuery(sql);

//STEP 5: Extract data from result set

while(rs.next()){

//Retrieve by column name

int id = rs.getInt("id");

int age = rs.getInt("age");

String first = rs.getString("first");

String last = rs.getString("last");

//Display values

System.out.print("ID: " + id);

System.out.print(", Age: " + age);

System.out.print(", First: " + first);

System.out.println(", Last: " + last);

}

//STEP 6: Clean-up environment

rs.close();

stmt.close();

conn.close();

}catch(SQLException se){

//Handle errors for JDBC

se.printStackTrace();

}catch(Exception e){

//Handle errors for Class.forName

e.printStackTrace();

}finally{

//finally block used to close resources

try{

if(stmt!=null)

stmt.close();

}catch(SQLException se2){

}// nothing we can do

try{

if(conn!=null)

conn.close();

}catch(SQLException se){

se.printStackTrace();

}//end finally try

}//end try

System.out.println("Goodbye!");

}//end main

}//end FirstExample

**Practical 7**

**Install TOMCAT web server and APACHE. Access the above developed static web pages for books website, using these servers by putting the web pages developed.**

**Theory:**

**Set the JAVA\_HOME Variable**

You must set the JAVA\_HOME environment variable to tell Tomcat where to find Java. Failing

to properly set this variable prevents Tomcat from handling JSP pages. This variable should list

the base JDK installation directory, not the bin subdirectory.

On Windows XP, you could also go to the Start menu, select Control Panel, chooseSystem, click

on the Advanced tab, press the Environment Variables button at the bottom, and enter the

JAVA\_HOME variable and value directly as:

Name: JAVA\_HOME

Value: C:\jdk

**Set the CLASSPATH**

Since servlets and JSP are not part of the Java 2 platform, standard edition, youhave to identify

the servlet classes to the compiler. The server already knows about theservlet classes, but the

compiler (i.e., javac ) you use for development probably doesn't.So, if you don't set your

CLASSPATH, attempts to compile servlets, tag libraries, or other classes that use the servlet and

JSP APIs will fail with error messages about unknown classes.

Name: JAVA\_HOME

Value: install\_dir/common/lib/servlet-api.jar

**Turn on Servlet Reloading**

The next step is to tell Tomcat to check the modification dates of the class files of requested servlets and reload ones that have changed since they were loaded into the server's memory. This slightly degrades performance in deployment situations, so is turned off by default. However, if you fail to turn it on for your development server,you'll have to restart the server every time you recompile a servlet that has already been loaded into the server's memory.

To turn on servlet reloading, edit install\_dir/conf/server.xml and add a DefaultContext subelement to the main Host element and supply true for the reloadable attribute. For example, in Tomcat 5.0.27, search for this entry:

<Host name="localhost" debug="0" appBase="webapps" ...>

and then insert the following immediately below it:

<DefaultContext reloadable="true"/>

Be sure to make a backup copy of server.xm before making the above change.

**Enable the Invoker Servlet:**

The invoker servlet lets you run servlets without first making changes to your Web application deployment descriptor. Instead, you just drop your servlet into WEB-INF/classes and use the URL

To enable the invoker servlet, uncomment the following servlet and servlet-mapping elements in

install\_dir/conf/web.xml. Finally, remember to make a backup copy of the original version of this

file before you make the changes.

<servlet>

<servlet-name>invoker</servlet-name>

<servlet-class>

org.apache.catalina.servlets.InvokerServlet

</servlet-class>

…

</servlet>

…

<servlet-mapping>

<servlet-name>invoker</servlet-name>

<url-pattern>/servlet/\*</url-pattern>

</servlet-mapping>

**Practical 8**

**Assume four users user1, user2, user3 and user4 having the passwords pwd1,pwd2, pwd3 and pwd4 respectively. Write a servlet for doing the following.1. Create a Cookie and add these four user id’s and passwords to this Cookie.2. Read the user id and passwords entered in the Login form (week1) and authenticate with the values (user id and passwords) available in the cookies.**

**Theory:**

**Servlet Life cycle:**

1. Servlet class loading

2. Servlet Instantiation

3. call the init method

4. call the service method

5. call destroy method

**Class loading and instantiation:**

If you consider a servlet to be just like any other Java program, except that it runs within a servlet container, there has to be a process of loading the class and making it ready for requests. Servlets do not have the exact equivalent of a main method that causes them to start execution.When a web container starts it searches for the deployment descriptor (WEB.XML) for each of its web applications. When it finds a servlet in the descriptor it will create an instance of the servlet class. At this point the class is considered to be loaded (but not initialised).

**The init method:**

The HttpServlet class inherits the init method from GenericServlet. The init method performs a role slightly similar to a constructor in an “ordinary” Java program in that it allows initialization of an instance at start up. It is called automatically by the servlet container and as it causes the application context (WEB.XML) to be parsed and any initialization will be performed. It comes in two versions, one with a zero parameter constructor and one that takes a ServletConfig parameter. The servlet engine creates a request object and a response object. The servlet engine invokes the servlet service() method, passing the request and response objects. Once the init method returns the servlet is said to be placed into service. The process of using init to initialize servlets means that it is possible to change configuration details by modifying the deployment descriptor without having them hard coded in with your Java source and needing a re-compilation.

void init(ServletConfig sc)

**Calling the service method:**

The service() method gets information about the request from the request object, processes the request, and uses methods of the response object to create the client response. The service method can invoke other methods to process the request, such as doGet(), doPost(), or methods you write. The service method is called for each request processed and is not normally overridden by the programmer.

The code that makes a servlet “go” is the. servlet

void service(ServletRequest req,ServletResponse res)

**The destroy Method:**

Two typical reasons for the destroy method being called are if the container is shutting down or if the container is low on resources. This can happen when the container keeps a pool of instances of servlets to ensure adequate performance. If no requests have come in for a particular servlet for a while it may destroy it to ensure resources are available for the servlets that are

being requested. The destroy method is called only once, before a servlet is unloaded and thus you cannot be certain when and if it is called.

void destroy()

**ServletConfig Class:**

The ServletConfig object is used by the Servlet Container to pass information to the Servlet during its initialization. Servlet can obtain information regarding initialization parameters and their values using different methods of ServletConfig class initialization parameters are name/value pairs used to provide basic information to the Servlet during its initialization like JDBC driver name, path to database, username, password etc.

**Methods of ServletConfig class:**

Following are the four methods of this class :

1. getInitParameter(String paramName) Returns value of the given parameter. If the value of parameter could not be found in the web.xml file then a null value is returned.

2. GetInitParameterNames() Returns an Enumeration object containing all the names of initialization parameters provided for this Servlet.

3. GetServletContext() Returns reference to the ServletContext object for this Servlet. It is similar to getServletContext() method provided by

HttpServlet class.

4. GetServletName() Returns name of the Servlet as provided in the web.xml file or if none is provided then returns the complete class path to the Servlet.

**Practical 9**

**Install a database (Mysql or Oracle). Create a table which should contain at least the following fields: name, password, email-id, phone number Write a java program/servlet/JSP to connect to that database and extract data from the tables and display them. Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page.**

**Code:**

CREATE TABLE `users` (

`user\_name` VARCHAR(100) NOT NULL,

`password` VARCHAR(100) NOT NULL,

`phone\_number` NUMBER(100) NOT NULL,

`email\_address` VARCHAR(50) NOT NULL,

PRIMARY KEY (`user\_name`)

)

COLLATE='latin1\_swedish\_ci'

ENGINE=InnoDB

**OUTPUT:**



**Practical 10**

**Write a JSP which inserts the details of the 3 or 4 users who register with the web site by using registration form. Authenticate the user when he submits the login form using the username and password from the database.**

**Code:**

index.jsp

<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head> <style> body

{ backgroundcolor:blue; color:black; }

</style>

<meta charset="UTF-8">

<title>Welcome to Login portal</title>

</head>

<body>

<center>

<h1>Login Here !</h1>

<form action="login.jsp" method="post"> Username:<input type="text" name="email"/><br/><br/>

Password:<input type="password" name="password"/><br/><br/>

<input type="submit" value="login"/>

<br>

<br>

<a style="color:blue" href="reg.jsp">Register for new user</a>

</center>

</form>

</body>

</html>

Login.jsp

<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF8" %>

<%@page import="java.sql.\*"%>

<html>

<style> body

{ background-color:blue; color:black; }

</style>

<%

String userid = request.getParameter("email"); String pwd = request.getParameter("password"); Class.forName("com.mysql.jdbc.Driver");

26

Connection con= DriverManager.getConnection ("jdbc:mysql ://localhost:3306 /practical14" ,"root","abc");

Statement st = con.createStatement(); ResultSet rs; rs = st.executeQuery("select \* from login where uname='" + userid

+ "' and pass='" + pwd

+ "'"); if (rs.next()) { session.setAttribute("userid", userid); out.println("<h1><center>welcome " + userid+"</h1></center><br><br>"); out.println("<h3><center>You are successfully logged in.</h3></center><br><br>"); out.println("<center><a style=\"color:black\"href=\"index.jsp\">Logout</a></center>");

} else {

out.println("<center><h1>Invalid password </h1><br><br> <a style=\"color:black\" href='index.jsp'>try again</a></center>");

}

%>

</html>

Reg.jsp

<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF- 8"%>

<!DOCTYPE html>

<html>

<head>

<style> body { background-color:blue; color:black;

}

</style>

<meta charset="UTF-8">

<title>Registration</title>

</head>

<body>

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

<center>

<table border="1" width="30%" cellpadding="5">

<thead>

<tr>

<th colspan="2">Registration form</th>

</tr>

</thead>

<tbody>

<tr>

<td>First Name</td>

<td><input type="text" name="fname" value="" /></td>

</tr>

<tr>

<td>Last Name</td>

<td><input type="text" name="lname" value="" /></td>

</tr>

<tr>

<td>Email</td>

<td><input type="text" name="email" value="" /></td>

</tr>

<tr>

<td>User Name</td>

<td><input type="text" name="uname" value="" /></td>

</tr>

<tr>

<td>Password</td>

<td><input type="password" name="pass" value="" /></td>

</tr>

<tr>

<td><input type="submit" value="Submit" /></td>

<td><input type="reset" value="Reset" /></td>

**Output:**

