PRACTICAL N0. 1

Aim: Write an HTML code to display your CV on a web page..

```
<html>
<head>
<title>RESUME | JOHN DOE</title>
</head>
<body>
<!-- BEGIN DIV FOR OVERALL BOX -->
<div id="resume">
<!-- THIS DIV CENTERS OUR HEADING -->
<h1>John Doe</h1>
<h2>4242 Ghila Road</h2>
<h2>Tucson, AZ 85701</h2>
<br />
<!-- END CENTERING DIV -->
</div>
<h2>Profile</h2>
<p>
Desires a resident position in the Bastyr University Acupuncture and Oriental
Medicine Residency Program. Able to be effective in a practice of any size.
Draw on experience with a range of patient issues, including additional work
in women and children’s care. Interested in health education for homeless.
Strong desire to contribute to the success of a program through an ability to
initiate and maintain relationships. Creative developer and presenter of
educational information.
</p>
<br />
<h2>Education</h2>
<h3>Masters of Acupuncture and Oriental Medicine,
<br />
Graduating June 2003</h3>
<p>
Bastyr University, Kenmore, WA 1999
</p>
```

```
<ul>
<li>
Completing an accredited program of coursework and supervised practice
in Acupuncture and Oriental Medicine. Extensive exposure to issues involving
women and children.
</li>
</ul>
<h3>Research Project</h3>
<ul>
<li>
Assisted the primary investigator in a double blind, randomized controlled
trail conducted at the Bastyr Center for Natural Health that evaluated the
effectiveness of Chinese herbs towards the control of Diabetes Mellitus in
post-menopausal women. Co-authored the research report that has been
submitted for publication to the Journal of Traditional Chinese Medicine.
</li>
</ul>
<h3>Bachelor of Science, Zoology</h3>
<p>
Miami University, Oxford, OH 1991 - 1995
</p>
<ul>
<li>
Participated in a community service project to increase citizen participation
in a clean up campaign.
</li>
<li>
Served as project leader in a fund raising project sponsored by the University
Student Council towards helping homeless youthsâ€™ return back to school.
</li>
</ul>
<br />
<h2>Related Experience</h2>
<h3>Bastyr University, Kenmore, WA 1999-present</h3>
<p>
Teaching Assistant
```

```
</p>
<ul>
<li>
Assists professor in the Anatomy & Physiology class.
</li>
<li>
Answers questions and demonstrate as needed.
</li>
</ul>
<h3>Kenmore Youth Ministry, Kenmore, WA 2000-2001</h3>
<p>
Camp Group Leader
</p>
<ul>
<li>
Participated in community youth group activities.
</li>
<li>
Developed activity programs now utilized by the youth ministry in helping
children improve reading skills.
</li>
</ul>
<h3>Franciscan Care Center Nursing Home,
<br />
Seattle, WA 1999 - 2000 </h3>
<p>
Volunteer Recreation Worker
</p>
<ul>
<li>
Provided social support to patients by reading to them, writing letters,
and visiting with them.
</li>
<li>
Formed friendships which enriched lives of patients
</li>
```

```html
</ul>
<!-- THIS DIV CENTERS OUR LINKS -->
<div id="bottom">
<p>
<a href="index.html">RESUME HOME</a> | <a href="#">SIMPLE RESUME</a> | <a href="resume.html">COMPLEX RESUME</a> | <a href="code.html" target="_blank">SEE HTML</a> | <a href="resume.css" target="_blank">SEE CSS</a>
</p>
</div>
<!-- END CENTERING LINKS -->
<!-- END DIV FOR OVERALL BOX -->
</div>
</body>
</html>
```

PRACTICAL NO. 2

Aim: Write an HTML code to create your Institute website, Department Website and Tutorial website for specific subject.

```
<html>
<head>
<title>Institute of Engineering</title>
</head>
<body>
<ul>
<li><a href="http://my.wm.edu">myWM</a></li>
<li><a href="http://directory.wm.edu/people/">Directory</a></li>
<li><a href="http://events.wm.edu">Events</a></li>
<li><a href="/about/visiting">Visit</a></li>
<li class="last-side-tactical"><a href="/atoz" id="wm-az">W&amp;M A-Z</a></li>
</ul>
</nav>
<nav id="side_top_nav">
<ul>
<li ><a href="/about" id="about" title="About William &#38; Mary">About</a></li>
<li ><a href="/academics" id="academics"
title="InstituteAcademics">Academics</a></li>
<li ><a href="/admission" id="admission" title="InstituteAdmission &#38;
Aid">Admission &#38; Aid</a></li>
<li ><a href="/research" id="research" title="InstituteResearch">Research</a></li>
<li ><a href="/campuslife" id="campus" title="InstituteCampus Life">Campus
Life</a></li>
<li ><a href="/news" id="news_events" title="InstituteNews">News</a></li>
<li><a href="http://www.tribeathletics.com/" id="athletics"
title="InstituteAthletics">Athletics</a></li>
<li><a href="http://www.wmalumni.com/" id="alumni"
```

```
title="InstituteAlumni">Alumni</a></li>
<li><a href="/giving" id="giving" title="InstituteGiving">Giving</a></li>
</ul>
</nav>
</div>
<header id="main-header"><div id="topbar"><a class="action_left icon-font" href="#"
id="menu_btn">&#xe9bd;</a><a class="mobile-header-logo" href="index.php"><img
alt="William and Mary" id="top_mark"
src="img/wm_wordmark_single_line_green.png"/></a><a class="action_right icon-font" href="#" id="search_btn">&#xe986;</a></div>
<div id="desktop_header">
<nav id="tactical_nav">
<ul>
<li>
<a href="http://my.wm.edu">myWM</a>
</li>
<li>
<a href="http://directory.wm.edu/people/">Directory</a>
</li>
<li>
<a href="http://events.wm.edu">Events</a>
</li>
<li>
<a href="/about/visiting">Visit</a>
</li>
<li>
<a href="/atoz" id="wm-az">W&amp;M A-Z</a>
</li>
</ul>
<
</nav>
```

```
<nav id="global_nav"><ul><li><a href="/about" id="about" >About</a>
</li><li><a href="/academics" id="academics" >Academics</a>
</li><li><a href="/admission" id="admission" >Admission &amp; Aid</a>
</li><li><a href="/research" id="research" >Research</a>
</li><li><a href="/campuslife" id="campus" >Campus life</a>
</li><li><a href="/news" id="news_events" >News</a>
</li><li><a href="http://www.tribeathletics.com/" id="athletics">Athletics</a>
</li><li><a href="http://www.wmalumni.com/" id="alumni">Alumni</a>
</li><li><a href="/giving" id="giving" >Giving</a>
</li></ul>
</nav>
</div>
<!-- end desktop_header -->
</header>
<!-- end main_header -->
<nav class="footer_col" id="audience_links">
<a href="/alumnigateway">Alumni</a>
<a href="/currentstudents">Current Students</a>
<a href="/employers">Employers</a>
<a href="/facultystaff">Faculty &amp; Staff</a>
<a href="/parentsandfamilies">Parents &amp; Families</a>
<a href="/friends">Friends &amp; Neighbors</a>
</nav>
<nav class="footer_col" id="additional_links">
<a href="http://swem.wm.edu">Library</a>
<a href="/offices/hr/careers">Careers at W&amp;M</a>
<a href="/offices/compliance/policies">Policies</a>
<a href="/about/administration/emergency">Emergency Information</a>
<a href="/aboutthissite">About this Site</a>
</nav>
<div id="contact_info">
<a.href-“/”class-“wordmark”>
```

```
<img alt="The College of William and Mary" id="footer_mark"
src="/img/wm_wordmark_single_line_white.png"/>
</a>
<p>Williamsburg, VA
<br/>
<a href="/contactus" class="contact-us">Contact Us</a>
All Rights Reserved &#169;
<span id="footercopyyear"></span>
</p>
</div>
</div>
</footer>
<div></div>
</body>
</html>
```

PRACTICAL NO. 3

Aim: Write an HTML program to design an entry form of student details and send it to store at 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 NO.4

Aim: Write programs using Java script for Web Page to display browsers information.

```
<html>
<head>
<title>Browser Information</title>
</head>
<body>
<h1>Browser Information</h1>
<hr>
<p>
The <b>navigator</b> object contains the following information
about the browser you are using.
</p>
<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>
```

PRACTICAL NO. 5

Aim: To design the scientific calculator and make event for each button using java script.

```
<body>
<div id=”big_wrapper”>
<h1 id=”heading”>SIMPLE SCIENTIFIC CALCULATOR</h1>
<div id=”form_wrapper”>
<form id=”formone” name=”calc”>
<input id=”display” type=”text” name=”display” value=” ” disabled contenteditable=”false” >
<br>
<input class=”button number” type=”button” value=”1” onClick=”calc.display.value+=1”>
<input class=”button number” type=”button” value=”2” onClick=”calc.display.value+=2”>
<input class=”button number” type=”button” value=”3” onClick=”calc.display.value+=3”>
<input class=”button three” type=”button” value=”C” onClick=”Resetfunction(this.form)”>
<input class=”button three” type=”button” value=”<-” onClick=”backspace(this.form)”>
<input class=”button three” type=”button” value=”=” onClick=”evaluation(this.form)”>
<br>
<input class=”button number” type=”button” value=”4” onClick=”calc.display.value+=4”>
<input class=”button number” type=”button” value=”5” onClick=”calc.display.value+=5”>
<input class=”button number” type=”button” value=”6” onClick=”calc.display.value+=6”>
<input class=”button opps” type=”button” value=”-” onClick=”calc.display.value+=’-‘”>
<input class=”button opps” type=”button” value=”%” onClick=”calc.display.value+=’%’”>
<input class=”button” type=”button” value=”cos” onClick=”cos_function()”>
<br>


<input class=”button number” type=”button” value=”7” onClick=”calc.display.value+=7”>
<input class=”button number” type=”button” value=”8” onClick=”calc.display.value+=8”>
<input class=”button number” type=”button” value=”9” onClick=”calc.display.value+=9”>
<input class=”button opps” type=”button” value=”*” onClick=”calc.display.value+=’*’”>
<input class=”button” type=”button” value=”n!” onClick=”fact_function()”>
<input class=”button” type=”button” value=”sin” onClick=”sin_function()”>
<br>
<input class=”button opps” type=”button” value=”.” onClick=”calc.display.value+=’.’”>
```

```
<input class=”button number” type=”button” value=”0” onClick=”calc.display.value+=0”>
<input class=”button opps” type=”button” value=”,” onClick=”calc.display.value+=’,’”>
<input class=”button opps” type=”button” value=”+” onClick=”calc.display.value+=’+’”>
<input class=”button opps” type=”button” value=”/” onClick=”calc.display.value+=’/’”>
<input class=”button” type=”button” value=”tan” onClick=”tan_function()”>
<br>
<input class=”button” type=”button” value=”E” onClick=”calc.display.value+=2.718”>
<input class=”button” type=”button” value=”pi” onClick=”calc.display.value+=3.141”>
<input class=”button” type=”button” value=”x^y” onClick=”power_function()”>
<input class=”button” type=”button” value=”(” onClick=”openpara(this.value)”>
<input class=”button” type=”button” value=”)” onClick=”closepara(this.value)”>
<input class=”button” type=”button” value=”log” onClick=”log_function()”>
<br>
<input class=”button” type=”button” value=”sqrt” onClick=”sqrt_function()”>
<input class=”button” type=”button” value=”LN2” onClick=”calc.display.value+=0.693”>
<input class=”button” type=”button” value=”LN10” onClick=”calc.display.value+=2.302”>
<input class=”button” type=”button” value=”log2E” onClick=”calc.display.value+=1.442”>
<input class=”button” type=”button” value=”log10E” onClick=”calc.display.value+=0.434”>
<input class=”button” type=”button” value=”EXP” onClick=”exp_function”>

</form>
</div>
</div>
</body>
```

PRACTICAL NO.7

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

Installing Tomcat on Windows can be done easily using the Windows installer. Its interface and functionality is similar to other wizard based installers, with only a few items of interest.

Installation as a service: Tomcat will be installed as a Windows service no matter what setting is selected. Using the checkbox on the component page sets the service as "auto" startup, so that Tomcat is automatically started when Windows starts. For optimal security, the service should be run as a separate user, with reduced permissions (see the Windows Services administration tool and its documentation).

Java location: The installer will provide a default JRE to use to run the service. The installer uses the registry to determine the base path of a Java 6 or later JRE, including the JRE installed as part of the full JDK. When running on a 64-bit operating system, the installer will first look for a 64-bit JRE and only look for a 32-bit JRE if a 64-bit JRE is not found. It is not mandatory to use the default JRE detected by the installer. Any installed Java 6 or later JRE (32-bit or 64-bit) may be used.

Tray icon: When Tomcat is run as a service, there will not be any tray icon present when Tomcat is running. Note that when choosing to run Tomcat at the end of installation, the tray icon will be used even if Tomcat was installed as a service.

Defaults: The defaults used by the installer may be overridden by use of the /C=<config file> command line argument. The configuration file uses the format name=value with each pair on a separate line. The names of the available configuration options are:

JavaHome

TomcatPortShutdown

TomcatPortHttp

TomcatPortAjp

TomcatMenuEntriesEnable

TomcatShortcutAllUsers

TomcatServiceDefaultName

TomcatServiceName

TomcatServiceFileName

TomcatServiceManagerFileName

TomcatAdminEnable

TomcatAdminUsername

TomcatAdminPassward

TomcatAdminRoles

By using /C=... along with /S and /D= it is possible to perform fully configured unattended installs of Apache Tomact.

Refer to the Windows Service HOW-TO for information on how to manage Tomcat as a Windows service.

The installer will create shortcuts allowing starting and configuring Tomcat. It is important to note that the Tomcat administration web application can only be used when Tomcat is running.

PRACTICAL NO.8

Aim: 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.

JDBC Driver Types

There are four types of JDBC drivers in use:

Type 1: JDBC-ODBC Bridge

A Type 1 JDBC-ODBC Bridge provides application developers with a way to access JDBC drivers via the JDBC API. Type 1 JDBC drivers translate the JDBC calls into ODBC calls and then send the calls to the ODBC driver. Type 1 JDBC drivers are generally used when the database client libraries need to be loaded on every client machine.

Type 2: Native API/Partly Java Driver

A Type 2 Native API/Partly Java Driver is a partial Java driver because it converts JDBC calls into database specific calls. Type 2 Native API/Partly Java Driver communicates directly with the database server.

Type 3: Pure Java Driver

A Type 3 Pure Java Driver works in a three tiered architecture. The JDBC calls are passed via the network to the middle tier server. This server translates the calls to the database specific native interface to further request the server. JDBC drivers available from Simba are Type 3 drivers.

Type 4: Native Protocol Java Driver

The type 4 driver is written completely in Java and is hence platform independent. It is installed inside the Java Virtual Machine of the client. It provides better performance over the type 1 and 2 drivers as it does not have the overhead of conversion of calls into ODBC or database API calls. Unlike the type 3 drivers, it does not need associated software to work.A Type 4 Native Protocol Java Driver converts JDBC calls into the database specific calls so that the client applications can communicate directly with the server.

PROGRAM:

Registration.html:

```
<html>
<head>
<title>Registration page</title>
</head>
<body bgcolor="#00FFFf">
<form METHOD="POST" ACTION="register">
<CENTER>
<table>
```

```
<center>
<tr> <td> Username </td>
<td><input type="text" name="usr"> </td> </tr>
<tr><td> Password </td>
<td><input type="password" name="pwd"> </td> </tr>
<tr><td>Age</td>
<td><input type="text" name="age"> </td> </tr>
<tr> <td>Address</td>
<td> <input type="text" name="add"> </td> </tr>
<tr> <td>email</td>
<td> <input type="text" name="mail"> </td> </tr>
<tr> <td>Phone</td>
<td> <input type="text" name="phone"> </td> </tr>
<tr> <td colspan=2 align=center> <input type="submit" value="submit"> </td> </tr>
</center>
</table>
</form>
</body>
```

Login.html

```
<html>
<head>
<title>Registration page</title>
</head>
<body bgcolor=pink> <center> <table>
<form METHOD="POST" ACTION="authent">
<tr> <td> Username </td>
<td><input type="text" name="usr"></td> </tr>
<tr> <td> Password </td>
<td> <input type="password" name="pwd"> </td> </tr>
<tr> <td align=center colspan="2"><input type="submit" value="submit"></td> </tr>
</table> </center>
</form>
</body>
</html>
```

Ini.java:

```
import javax.servlet.*;
import java.sql.*;
import java.io.*;
public class Ini extends GenericServlet
{
private String user1,pwd1,email1;

public void service(ServletRequest req,ServletResponse res) throws ServletException,IOException
{
user1=req.getParameter("user");
pwd1=req.getParameter("pwd");
email1=req.getParameter("email");
res.setContentType("text/html");
PrintWriter out=res.getWriter();
try
   {
    Class.forName("oracle.jdbc.driver.OracleDriver");
    Connection
con=DriverManager.getConnection("jdbc:oracle:thin:@195.100.101.158:1521:cclab","scott","tiger");
   PreparedStatement st=con.prepareStatement("insert into personal values(?,?,?,?,?,?)");
 st.setString(1,user1);
st.setString(2,pwd1);
st.setString(3,"25");
st.setString(4,"hyd");
st.setString(5,email1);
st.setString(6,"21234");
st.executeUpdate();
con.close();
  }
catch(SQLException s)
{ out.println("not found "+s);
}
catch(ClassNotFoundException c)
{
   out.println("not found "+c);
}

} }
```

web.xml:

```
<web-app>
<servlet>
<servlet-name>init1</servlet-name>
<servlet-class>Ini</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>init1</servlet-name>
<url-pattern>/regis</url-pattern>
</servlet-mapping>
</web-app>
```

PRACTICAL NO.9

Aim: Write a JSP which insert 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 user name and password from the database

Java Server Pages (JSP) is a technology that helpssoftware developers create dynamically generated web pages based on HTML, XML, or other document types. JSP pages use several delimiters for scripting functions. The most basic is , which encloses a JSP scriptlet. A scriptlet is a fragment of Java code that is run when the user requests the page. Other common delimiters include for expressions, where the value of the expression is placed into the page delivered to the user, and directives, denoted with JDBC is a Java-based data access technology from Sun Microsystems. It is an acronym as it is unofficially referred to as Java Database Connectivity. This technology is an API for the Java programming language that defines how a client may access adatabase. It provides methods for querying and updating data in a database. JDBC is oriented towards relational databases. A JDBC driver is a software component enabling a Java application to interact with a database. To connect with individual databases, JDBC (the Java Database Connectivity API) requires drivers for each database. The JDBC driver gives out the connection to the database and implements the protocolfor transferring the query and result between client and database.

PROGRAM: Login.jsp:

```
<%@ page import="java.sql.*;" %>
<html>
<body>
<%
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection
con=DriverManager.getConnection("jdbc:odbc:web",
"","");
PreparedStatement st=con.prepareStatement("select
password from users where
username=?");
String s1=request.getParameter("t1");
st.setString(1,s1);
ResultSet rs=st.executeQuery();
if(rs.next())
{
String s2=request.getParameter("t2");
if(s2.equals(rs.getString(1)))
out.println("<h2
style='color:blue;font-
family:cambria;textalign:center'>Welcome "+s1+"
!!!</h2>");
else
out.println("<h3 style='color:blue;font-
family:cambria;textalign:center">Invalid
Password!!! Please Enter Correct
Password</h3>");
}
else
out.println("<h3
style='color:blue;font-
family:cambria;text-
align:center">Invalid User!!!
Please Register</h3>");
rs.close();
st.close();
con.close();
}
catch(Exception e)
{
out.println(e);
}
```