**NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY** 

An Autunomous Institution Approved y UGC/ AICTE/Govt. of Karnataka ,Accredited by NBA(Tier-I) andNAAC ‘A+’ Grade

Affiliated to Visvesvaraya Technological University, Belagavi, Yelahanka, Bengaluru-64



WEB PROGRAMMING

LAB MANUAL 2022-23| SCHEME-2022| 2ND SEMESTER

DEPARTMENT OF mca

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Web Programming Laboratory**  [As per Choice Based Credit System (CBCS) scheme]  **SEMESTER – II** | | | | |
| Subject Code | | 22MCA208L | CIE Marks | 50 |
| Number of Hours/Week | | 02 Hrs. Laboratory | SEE Marks | 50 |
|  | |  | SEE Hours | 03 |
| **CREDITS – 01** | | | | |
| **1** | Write a HTML Page to create a Student registration form that collects various student information such as first name, last name, Roll No., email id, mobile number, etc and a two buttons: submit and reset. Apply different CSS properties. | | | |
| **2** | 1. Develop and demonstrate a XHTML file that includes JavaScript script for the following problems:    1. Accept a number n obtained using prompt and display the first n Fibonacci numbers using alert    2. Accept a number n obtained using prompt, and display a table of numbers from 1 to n and their squares using alert() 2. Develop and demonstrate using JavaScript, a XHTML document that displays random numbers (integers). | | | |
| **3** | Write a JavaScript program to generate n number of random numbers and store them in an array. Sort the generated numbers in ascending order using array sort method. Develop separate functions to find mean and median of numbers that are in the array. Display the results with appropriate messages. | | | |
| **4** | 1. Develop and demonstrate, using JavaScript script, a XHTML document that collects the USN (the valid format is: A digit from 1 to 4 followed by two uppercase characters followed by two digits followed by two uppercase characters followed by three digits; No embedded spaces allowed) of the user. Event handler must be included for the Form element that collects this information to validate the input. Messages in the alert windows must be produced when errors are detected. 2. Modify the above program to get the current semester also (restricted to be a number from 1 to 6). | | | |
| **5** | Develop using JavaScript script, an XHTML document that use of onload and onfocus events. | | | |
| **6** | 1. Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, and Name of the College, Brach, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document. 2. Create an XSLT style sheet for one student element of the above document and use it to create a display of that element. | | | |
| **7** | 1. Create XHTML forms with Name, address line1, address line2 and email text fields. On submitting, store the values in MySQL table. Retrieve and display the data based on name. 2. Implement a PHP program to read student data from an XML file and store into the MySQL database. Retrieve and display using SEARCH function. | | | |
| **8** | Implement the following web applications using AngularJS: A simple calculator web application that takes two numbers and an operator (+, -, /, \* and %) from an HTML page and returns the result page with the operation performed on the operands | | | |
| **9** | Implement the following web applications using AngularJS: A user validation web application, where the user submits the login name and password to the server. The name and password are checked if the data matches, a successful login page is returned. Otherwise a failure message is shown to the user. | | | |
| **10** | Implement the following web applications using PHP: A user validation web application, where the user submits the login name and password to the server. The name and password are checked against the data already available in Database and if the data matches, a successful login page is returned. Otherwise a failure message is shown to the user. | | | |
| **ASSESSMENT METHODS**   |  |  | | --- | --- | | Experiment Write up + Execution + Viva | 20 | | Lab Record Writing | 10 | | Lab Internals Test | 10 | | Mini Project | 10 | | Total | 50 |  * Final Exam will be conducted for 50 marks (SEE) | | | | |
| **Course Outcome (CO): At the end of this course, the students will be able to:**  **CO1:** Use HTML and CSS to design web pages.  **CO2:** Infer the role of XML in web applications and use them.  **CO3:** Apply the concepts of JavaScript to perform client side validation and create dynamic web pages.  **CO4:** Create server side applications using PHP.  **CO5:** Create web application using AngularJS. | | | | |

|  |
| --- |
| **1. Write a HTML Page to create a Student registration form that collects various student information such as first name, last name, Roll No., email id, mobile number, etc and a two buttons: submit and reset. Apply different CSS properties.**    <html>  <head> <title> Registration Page </title> </head>  <body >  <style>  form{  margin-top:10px;  margin-bottom: 300px;  margin-right: 350px;  margin-left: 350px;  background: Lightskyblue ;  border: 4px solid #000000;  }  </style>  <form>  <h1 align="center"> Student Registration Form</h1>  <label> Firstname </label>  <input type="text" name="firstname" size="15"/> <br> <br>  <label> Middlename: </label>  <input type="text" name="middlename" size="15"/> <br> <br>  <label> Lastname: </label>  <input type="text" name="lastname" size="15"/> <br> <br>  <label> Course : </label>  <select>  <option value="Course">Course</option>  <option value="BCA">BCA</option>  <option value="BBA">BBA</option>  <option value="MBA">MBA</option>  <option value="MCA">MCA</option>  </select> <br> <br>  <label> Gender : </label><br>  <input type="radio" name="male"/> Male <br>  <input type="radio" name="female"/> Female <br> <br> <br>  <label> Phone : </label>  <input type="text" name="country code" value="+91" size="2"/>  <input type="text" name="phone" size="10"/> <br> <br>  Address <br>  <textarea cols="60" rows="5" value="address"></textarea><br> <br >  Email:  <input type="email" id="email" name="email"/> <br>  <br> <br>  Password:  <input type="Password" id="pass" name="pass"> <br><br>  <input type="button" value="Submit"/>  <input type="button" value="Reset"/>  </form>  </body>  </html>  **OUTPUT:**    **2a. Develop and demonstrate a XHTML file that includes JavaScript script for the following problems:**   * 1. **Accept a number n obtained using prompt and display the first n Fibonacci numbers using alert**   <html>  <head><title> Exercise 2a.i</title></head>  <body>  <script type="text/javascript">  var fib1=0,fib2=1,fib=0;  var num = prompt("Enter a number : \n", "");  if(num!=null && num>0)  {  document.write("<h1>" + num + " Fibonocci are <br></h1>");  if(num==1)  document.write("<h1> "+ fib1 + "</h1>");  else  document.write("<h1>" + fib1 + "<br/> " + fib2 + "</h1>");  for(i=3;i<=num; i++)  {  fib= fib1 + fib2;  document.write("<h1> " + fib + "</h1>");  fib1=fib2;  fib2=fib;  }  }  else  alert("No Proper Input");  </script>  </body>  </html>  **OUTPUT:**    **ii) Input: A number n obtained using prompt**  **Output: A table of numbers from 1 to n and their squares using alert**  <!DOCTYPE html>  <head><title> Exercise 2a.ii </title></head>  <body>  <script type="text/javascript">  var n=prompt("Enter positive value for n: "," ");  while(n<=0)  {  alert("Enter positive value");  n=prompt("Enter positive value for n: "," ");  }  document.write("Numbers and their square values displayed using alert ..<br />");  for(i=1;i<=n;i++)  {  document.write("The Square of " + i + " is " + (i\*i) + "<br/>");  alert("The Square of " + i + " is " + (i\*i) + "<br/>");  }  </script>  </body>  </html>  **OUTPUT:**    Numbers and their square values displayed using alert. The Square of 1 is 1 The Square of 2 is 4 The Square of 3 is 9 The Square of 4 is 16  **2b) Develop and demonstrate using JavaScript, a XHTML document that displays random numbers (integers).**  <!DOCTYPE html>  <body>  <p id="demo">Click the button to display a random number.</p>  <button onclick="myFunction()">Try it</button>  <script>  function myFunction()  {  document.getElementById("demo").innerHTML=parseInt(Math.random()\*100);  }  </script>  </body>  </html>  **OUTPUT:**      **3.** **Write a JavaScript program to generate n number of random numbers and store them in an array. Sort the generated numbers in ascending order using array sort method. Develop separate functions to find mean and median of numbers that are in the array. Display the results with appropriate messages.**  <html>  <body>  <script>  function genRanNum(n) {  const randomNumbers = Array.from({length: n},()=> Math.floor(Math.random() n));  return randomNumbers;  }  function findMean(numbers) {  const sum = numbers.reduce((acc, num) => acc + num, 0);  return sum / numbers.length;  }  function findMedian(numbers) {  const sortedNumbers = numbers.slice().sort((a, b) => a - b);  const middleIndex = Math.floor(sortedNumbers.length / 2);  if (sortedNumbers.length % 2 == 0) {  return (sortedNumbers[middleIndex - 1] + sortedNumbers[middleIndex]) / 2;  } else {  return sortedNumbers[middleIndex];  }  }  const n = 10; // Change this to the desired number of random numbers  const randomNumbers = genRanNum(n);  const sortedNumbers = randomNumbers.slice().sort((a, b) => a - b);  const mean = findMean(sortedNumbers);  const median = findMedian(sortedNumbers);  document.write(`Generated Random Numbers: ${randomNumbers}`);  document.write("<br>");  document.write(`Sorted Numbers: ${sortedNumbers}`);  document.write("<br>");  document.write(`Mean: ${mean}`);  document.write("<br>");  document.write(`Median: ${median}`);  </script>  </body>  </html>  **4a) Develop and demonstrate, using JavaScript script, a XHTML document that collects the USN (the valid format is: A digit from 1 to 4 followed by two uppercase characters followed by two digits followed by two uppercase characters followed by three digits; No embedded spaces allowed) of the user. Event handler must be included for the Form element that collects this information to validate the input. Messages in the alert windows must be produced when errors are detected.**  **4b) Modify the above program to get the current semester also (restricted to be a number from 1 to 4).**  <html>  <head><title>Exercise 4a & b </title>  <script type="text/javascript">  function chkusn() {  var myusn=document.getElementById("usn");    var pos=myusn.value.search(/^[1-4][A-Z][A-Z]\d{2}[A-Z][A-Z]\d{3}$/);  if(myusn.value== "")  {  alert("Enter the USN.");  myusn.focus();  return false;  }  if(pos != 0) {  alert("The format of the USN is not correct.");  myusn.focus();  myusn.select();  return false;  } else  return true;  }  function chksem() {  var mysem=document.getElementById("sem");  var pos=mysem.value.search(/^[1-4]$/);  if(mysem.value== "")  {  alert("Enter the SEM value.");  mysem.focus();  return false;  }  if(pos != 0) {  alert("Enter values 1 to 4 for semester.");  mysem.focus();  mysem.select();  return false;  } else  return true;  }  </script>  </head>  <body >  <h3> Validation Example </h3><br /><br />  <p> Example to check the format of the data entered in the text box </p><br /><br /><br />  <form action=" ">  <p> USN : <input type="text" id="usn" /><br /><br />  SEM : <input type="text" id="sem" /><br /><br /><br />  <input type="submit" id="submit" value="Submit" />  </p>  </form>  <script type="text/javascript">  document.getElementById("usn").onchange=chkusn;  document.getElementById("sem").onchange=chksem;  </script>  </body>  </html>  **OUTPUT:**    **5) Develop using JavaScript script, an XHTML document that use of onload and onfocus events**  <html>  <head><title>onfocus\_events</title>  <script type="text/javascript" >  function computecost()  {  var nb1 = document.getElementById("French").value;  var nb2 = document.getElementById("hazelnut").value;  var nb3 = document.getElementById("Columbian").value;  var nb4 = (nb1 \* 3.49) + (nb2 \* 5.40)+ (nb3 \* 6.20);  document.getElementById("cost").value = nb4;  }  </script>  </head>  <body onload="myFunction()">  <script>  function myFunction() {  alert("Onload Event!");  }  function color1(x) {  x.style.background = "yellow";  }  </script>  <form action=" ">  <h3>Coffee Order Form</h3>  <table border="border">  <tr>  <th>Product name</th>  <th>Price</th>  <th>Quantity</th>  </tr>  <tr>  <th>French vanilla</th>  <td>Rs.3.49</td>  <td><input type="text" id="French" size="2"/></td>  </tr>  <tr>  <th>Hazelnut cream</th>  <td>Rs.5.40</td>  <td><input type="text" id="hazelnut" size="2" onfocus="color1(this)"/></td>  </tr><tr>  <th>Columbian</th>  <td>Rs.6.20</td>  <td><input type="text" id="Columbian" size="2"/></td>  </tr>  </table>  <p>  <input type="button" value="Total cost" onmousemove="computecost()";/>  <input text="text" size="5" id="cost" onfocus="this.blur()";/>  </p>  <p>  <input type="button" value="Submit order" onClick="computecost()";/>  <input type="reset" value="Reset order"/>  </p>  </form>  </body>  </html>  **OUTPUT:**    **6a) Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, Name of the College, Brach, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.**  <?xml version="1.0" encoding="utf-8"?>  <?xml-stylesheet href="Info.css" type="text/css"?>  <student>  <stud-info>Student Information</stud-info>  <stud1>  <usn>USN: 1NT22MC001</usn>  <name>Name: Aashish S</name>  <noc>College: NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY</noc>  <branch>Branch: MCA</branch>  <yoj>Year: 2022</yoj>  <eid>Email: aashishh@me.com</eid>  </stud1>  <br/>  <stud2>  <usn>USN: 1NT22MC002</usn>  <name>Name: Anish L R</name>  <noc>College: NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY</noc>  <branch>Branch: MCA</branch>  <yoj>Year: 2022</yoj>  <eid>Email: anishh@me.com</eid>  </stud2>  <br/>  <stud3>  <usn>USN: 1NT22MC003</usn>  <name>Name: Amal Antony</name>  <noc>College: NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY</noc>  <branch>Branch: MCA</branch>  <yoj>Year: 2022</yoj>  <eid>Email: amal@me.com</eid>  </stud3>  </student>  **CSS**  stud-info{display:block; color:blue; font-style:italic; font-weight:bold; font-size: 200%;}  student{display:block; font-size:100%; margin:200px;border: 1px solid silver; padding:0.5em;}  stud1 { display:block; color:blue; margin:0.5em;}  stud2 { display:block; color:red; margin:0.5em;}  stud3 { display:block; color:pink; margin:0.5em;}  usn,name,noc,branch,yoj,eid { display:block; margin:0.5em;}  **OUTPUT:**    **6b) Create an XSLT style sheet for one student element of the above document and use it to create a display of that element.**  <?xml version = "1.0"?>  <?xml-stylesheet type = "text/xsl" href = "6b.xslt" ?>  <students>  <VTU>  <USN> 1RL01CS012 </USN>  <name> GuruPrasad </name>  <college> RLJIT </college>  <branch> CSE</branch>  <YOJ> 2006 </YOJ>  <email> gp@gmail.com </email>  </VTU>  <VTU>  <USN> 1Rl06CS053</USN>  <name> SukruthGowda</name>  <college> RLJIT </college>  <branch>CSE </branch>  <YOJ>2006</YOJ>  <email>sukruth@gmail.com</email>  </VTU>  <VTU>  <USN> 1RL06CS001</USN>  <name>Abhishekbose</name>  <college> RLJIT</college>  <branch> CSE </branch>  <YOJ>2006</YOJ>  <email>abhishek@yahoo.com </email>  </VTU>  </students>  **XSLT**  <?xml version = "1.0"?>  <xsl:stylesheet version = "1.0" xmlns:xsl = "http://www.w3.org/1999/XSL/Transform" xmlns = "http://www.w3.org/1999/xhtml" >  <xsl:template match = "students">  <h2> VTU Students' Descriptions </h2>  <xsl:for-each select = "VTU">  <span style = "font-style: italic; color: blue;"> USN: </span>  <xsl:value-of select = "USN" /> <br />  <span style = "font-style: italic; color: blue;"> Name: </span>  <xsl:value-of select = "name" /> <br />  <span style = "font-style: italic; color: green;"> College: </span>  <xsl:value-of select = "college" /> <br />  <span style = "font-style: italic; color: red;"> Branch: </span>  <xsl:value-of select = "branch" /> <br />  <span style = "font-style: italic; color: yellow;"> Year of Join: </span>  <xsl:value-of select = "YOJ" /> <br />  <span style = "font-style: italic; color: blue;"> E-Mail: </span>  <xsl:value-of select = "email" /> <br /> <br />  </xsl:for-each>  </xsl:template>  </xsl:stylesheet>  **OUTPUT:**    **7a) Create XHTML form with Name, address line1, address line2 and email text fields. On submitting, store the values in MySQL table. Retrieve and display the data based on name.**   * **Save the below program in C:\xampp\htdocs as guestbook.php** * **Run the XAMPP server**   <?php  $dbname = "guestbook";  $servername = "localhost";  $username = "root";  $password = "";  $name = isset($\_POST['name'])?$\_POST['name']:'';  $add1 = isset($\_POST['add1'])?$\_POST['add1']:'';  $add2 = isset($\_POST['add2'])?$\_POST['add2']:'';  $email = isset($\_POST['email'])?$\_POST['email']:'';  $con = mysqli\_connect($servername,$username,$password,$dbname);  $query = "INSERT into guestinfo values('','$name','$add1','$add2','$email')";  if(isset($\_POST['submit'])) {  if(strlen($name == '' || $add1 == '' || $email == '')) {  echo "Sorry, feilds cannot be left blank";  } else {  $result = mysqli\_query($con,$query);  if($result) {  echo "Details successfully inserted";  } else {  echo "Sorry, something went wrong. Could not insert details ".mysqli\_error($con);  }  }    }    ?>  <?php  if(isset($\_POST['search'])) {  $key = isset($\_POST['keyword'])?$\_POST['keyword']:'';  $find\_name = mysqli\_query($con, "SELECT name,add1,add2,email FROM guestinfo WHERE name LIKE '%$key%' ");  $count = mysqli\_num\_rows($find\_name);  if($count == 0) {  echo "Sorry, No such name exist in the database..";  } else if($count == 1){  echo "<table border='1'><tr><th>Name</th><th>Address1</th><th>Email Address</th></tr>";  while ($row = mysqli\_fetch\_assoc($find\_name)) {  echo "<tr>  <td>$row[name]</td>  <td>$row[add1]</td>  <td>$row[email]</td>  </tr></table>";  }  }  }  ?>  <!DOCTYPE html>  <html>  <head>  <title>My GuestBook</title>  <!-- The styling goes here... -->  <style type="text/css">  body{  margin-left: 25%;  }  input {  width: 40%;  padding: 12px;  margin: 7px 1px;  }  textarea {  width: 40%;  padding: 12px;  margin: 7px 1px;  }  input[type=submit] {  width: 20%;  margin-left: 10%;  }  </style>  <!-- End of styling -->  </head>  <body>  <h2>Guest Book Applicatioin</h2>  <form action="guestbook.php" method="post">  <input type="text" name="name" placeholder ="Please enter your name"><br>  <textarea name="add1" placeholder="Enter Address1"></textarea> <br>  <textarea name="add2" placeholder="Enter Address2"></textarea><br>  <input type="text" name="email" placeholder="Enter email address"><br>  <input type="submit" name="submit" value="Send Info!!">  </form>  <br><br>  <h3>Student Name Search</h3>  <form action="guestbook.php" method="POST">  <input type="text" name="keyword" placeholder="Please enter the name of student"><br>  <input type="submit" name="search" value="Search Name">  </form>  </body>  </html>  **OUTPUT:**  **Open the browser and type the URL http://localhost/guestbook.html**      **7b) Write a PHP program to read student data from an XML file and store into the MySQL database. Retrieve and display using SEARCH function.**  **Save it in C:\xampp\htdocs as dbconnect.php**  <!-- Database connectivity-->  <?php        $dbname = "xml";      $servername = "localhost";      $username = "root";      $password = "";        $conn = mysqli\_connect($servername,$username,$password,$dbname);      if(!$conn) {          echo "Sorry unable to connect to database ".mysqli\_connect\_error();      }  ?>  <!-- Database connectivity Ends-->  **Save it in C:\xampp\htdocs as searchxml.html**  <?php include 'dbconnect.php'; ?>  <!DOCTYPE html>  <html>  <head>      <title>XML Data Search</title>      <!--  The styling goes here...  -->      <style type="text/css">          input {              width: 40%;              padding: 12px;              margin: 7px 1px;          }          input[type=submit] {              width: 20%;              margin-left: 10%;          }      </style>      <!--  End of styling  -->  </head>  <body>  <h2> Fetch Data</h2>  <form action="searchxml.php" method="POST">      <input type="text" name="search" placeholder="Enter the data to search">      <input type="submit" name="submit" value="search">  </form>  </body>  </html>  **Save it in C:\xampp\htdocs as readxml.php**  <?php include 'dbconnect.php' ; ?>  <!-- XML Load File function-->  <?php        $xml = simplexml\_load\_file('samplefile.xml');      foreach ($xml->student as $student) {          $id =  $student->id;          $name =  $student->name;          $age =  $student->age;          $query = mysqli\_query($conn, "INSERT INTO mca values('$name',$age')");      }  ?>  <!-- XML loadfile function Ends-->  **Save it in C:\xampp\htdocs as searchxml.php**  <?php include 'dbconnect.php' ; ?>  <?php  if(isset($\_POST['submit'])) {          $key = isset($\_POST['search'])?$\_POST['search']:'';          $find\_name = mysqli\_query($conn, "SELECT \* FROM mca WHERE name LIKE '%$key%'");          $count = mysqli\_num\_rows($find\_name);          if($count == 0) {              echo "Sorry, No such name exist in the database..";          } else if ($count==1){              echo "<table border='1' width='500'><tr><th>Name</th><th>Age</th></tr>";              while ($row = mysqli\_fetch\_assoc($find\_name)) {                  echo "<tr>                      <td>$row[name]</td>                      <td>$row[age]</td>                      </tr></table>";              }          } else{              echo "state 1";          }      }  $conn->close();  ?>  **save it in c:\xampp\htdocs as samplefile.xml**  <Students>  <student>  <name>daiwik</name>  <age>20</age>  </student>  <student>  <name>soham</name>  <age>22</age>  </student>    <student>  <name>himani</name>  <age>21</age>  </student>  <student>  <name>deetya</name>  <age>20</age>  </student>    </Students>  **OUTPUT:**   * **Run the XAMPP server.** * **Create the database.** * **Open the browser and execute the readxml.php** * **Open the browser and execute the searchxml.php, searchxml.html**     **8. Implement the following web applications using AngularJS: A simple calculator web application that takes two numbers and an operator (+, -, /, \* and %) from an HTML page and returns the result page with the operation performed on the operands**  <html>  <head>  <h2> Simple Calculator using AngularJS</h2>  </head>  <script src= "https://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/angular.min.js">  </script>  <style>  body{font: 12px/20px 'Lucida Grande' Tahoma, Verdana, sans-serif;  color: red;  text-align: center;  }  </style>  <script>  angular.module('CalculatorApp', []).controller('CalculatorController', function($scope) {  $scope.result = function() {  if ($scope.operator == '+') {  return $scope.a + $scope.b;  }  if ($scope.operator == '-') {  return $scope.a - $scope.b;  }  if ($scope.operator == '\*') {  return $scope.a \* $scope.b;  }  if ($scope.operator == '/') {  return $scope.a / $scope.b;  }  };  });  </script>  <body>  <div ng-app="CalculatorApp" ng-controller="CalculatorController">  <p><input type="number" ng-model="a"></p>  <p><input type="number" ng-model="b"></p>  <p><select ng-model="operator">  <option>+</option>  <option>\*</option>  <option>-</option>  <option>/</option>  </select></p>  <p>Result: {{ result() }}</p>  </div>  </body></html>  **OUTPUT:**    **9. Implement the following web applications using AngularJS: A user validation web application, where the user submits the login name and password to the server. The name and password are checked if the data matches, a successful login page is returned. Otherwise a failure message is shown to the user.**  <!DOCTYPE html>  <html ng-app="myApp">  <head>  <title>Login Page</title>  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>  <style>  body{font: 12px/20px 'Lucida Grande' Tahoma, Verdana, sans-serif;  text-align: center;  }  </style>  <script>  var app = angular.module('myApp', []);  app.controller('loginCtrl', function ($scope) {  $scope.login = function () {  var enteredUsername = $scope.username;  var enteredPassword = $scope.password;  if (enteredUsername === 'Admin' && enteredPassword === 'Password@123') {  $scope.message = 'Login Successful';  $scope.messageColor = 'green';  } else {  $scope.message = 'Invalid username or password';  $scope.messageColor = 'red';  }  };  });  </script>  </head>  <body>  <div ng-controller="loginCtrl">  <h1>Login Page</h1>  <form ng-submit="login()">  <label for="username">Username:</label>  <input type="text" id="username" ng-model="username" required><br><br>  <label for="password">Password:</label>  <input type="password" id="password" ng-model="password" required><br><br>  <button type="submit">Login</button>  </form>  <div ng-show="message" style="color: {{ messageColor }}">{{ message }}</div>  </div>  </div>  </body>  </html>  **OUTPUT:**    **10. Implement the following web applications using PHP: A user validation web application, where the user submits the login name and password to the server. The name and password are checked against the data already available in Database and if the data matches, a successful login page is returned. Otherwise a failure message is shown to the user.**  **database.php**  <?php  $host = "localhost";  $username = "root";  $password = "";  $database = "student";  $conn = new mysqli($host, $username, $password, $database);  if ($conn->connect\_error) {      die("Connection failed: " . $conn->connect\_error);  }  ?>  **login.html**  <!DOCTYPE html>  <html>  <head>      <title>Login</title>  </head>  <body>      <h2>Login</h2>      <form action="login.php" method="post">          <label for="username">Username:</label>          <input type="text" id="username" name="username" required><br><br>          <label for="password">Password:</label>          <input type="password" id="password" name="password" required><br><br>          <input type="submit" value="Login">      </form>  </body>  </html>  **login.php**  <?php  session\_start();  require\_once("database.php");  if ($\_SERVER["REQUEST\_METHOD"] == "POST") {      $username = $\_POST["username"];      $password = $\_POST["password"];      $stmt = $conn->prepare("SELECT  username, password FROM users WHERE username = ?");      $stmt->bind\_param("s", $username);      $stmt->execute();      $result = $stmt->get\_result();      if ($result->num\_rows == 1) {          $row = $result->fetch\_assoc();          if (password\_verify($password, $row["password"])) {              $\_SESSION["username"] = $row["username"];              header("Location: welcome.php");          } else {              echo "Login failed. Please try again.";          }      } else {          echo "Login failed. Please try again.";      }      $stmt->close();  }  $conn->close();  ?>  **register.html**  <!DOCTYPE html>  <html>  <head>      <title>Register</title>  </head>  <body>      <h2>Register</h2>      <form action="register.php" method="post">          <label for="username">Username:</label>          <input type="text" id="username" name="username" required><br><br>          <label for="password">Password:</label>          <input type="password" id="password" name="password" required><br><br>          <input type="submit" value="Register">      </form>  </body>  </html>  **register.php**  <?php  require\_once("database.php");  if ($\_SERVER["REQUEST\_METHOD"] == "POST") {      $username = $\_POST["username"];      $password = password\_hash($\_POST["password"], PASSWORD\_DEFAULT);      $stmt = $conn->prepare("INSERT INTO users (username, password) VALUES (?, ?)");      $stmt->bind\_param("ss", $username, $password);      if ($stmt->execute()) {          echo "Registration successful. <a href='login.html'>Login</a>";      } else {          echo "Registration failed. Please try again.";      }      $stmt->close();  }  $conn->close();  ?>  **welcome.php**  <?php  session\_start();  if (!isset($\_SESSION["username"])) {      header("Location: login.html");  }  echo "Welcome, " . $\_SESSION["username"] . "! You are successfully logged in.";  ?>  **OUTPUT:** |

**PART-B**

**Develop a web application project using the languages and concepts learnt in the theory and exercises listed in part A with a good look and feel effects. You can use any web technologies and frameworks and databases.**

Note:

1. In the examination each student picks one question from part A.

2. A team of two or three students must develop the mini project. However during the examination, each student must demonstrate the project individually.

3. The team must submit a brief project report (25-30 pages) that must include the following:

a. Introduction

b. Requirement Analysis

c. Software Requirement Specification

d. Analysis and Design

e. Implementation

f. Testing

4. The report must be evaluated for 10 Marks. Demonstration and Viva for 20 Marks.