DEPARTMENT OF COMPUTER ENGINEERING

LABORATORY MANUAL

WEB TECHNOLOGY LABORATORY

SEMINAR-II

(BTCOS407)

Shri Vile Parle Kelavani Mandal's
INSTITUTE OF TECHNOLOGY DHULE (M.S.)

DEPARTMENT OF COMPUTER ENGINEERING

Vision of Institute

To be a socially sensitive engineering institute of excellence adding value to the nation.

Mission of Institute

1. To provide resources of excellence with a focus on nurturing and developing the society.
2. To strive to be an institute of global recognition.

Vision of Department

We envision a globally recognized and innovative computer engineer who meets socio-economical and industrial needs.

Mission of Department

1.To empower students with comprehensive knowledge of Computer Engineering to be successful professionals.

2.To ensure quality education that prepares students for careers in industry and higher education.

3.To develop leadership skills in students while instilling strong ethical values and encouraging lifelong learning.

Program Educational Objectives (PEOs) of Department

PEO I: Graduate shall have successful professional careers, lead and manage teams.

PEO II: Graduate shall exhibit functional and disciplinary skills to resolve real life problems.

PEO III: Graduate shall evolve as a professional or researcher and continue to learn and adapt emerging technology.

COMPUTER ENGINEERING DEPARTMENT

Program Outcomes

Engineering Graduates will be able to:

1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

COMPUTER ENGINEERING DEPARTMENT

Program Specific Outcomes (PSO) addressed by the Course:

A graduate of the Computer Engineering Program will demonstrate-

PSO1: Professional Skills-The ability to understand, analyze and develop computer programs in the areas related to algorithms, system software, multimedia, web design, data science, and networking for efficient design of computer-based systems.

PSO2: Problem-Solving Skills- The ability to apply standard practices and strategies in software project development using open-ended programming environments to deliver advanced computing systems.

PSO3: Professional Career and Entrepreneurship- The ability to employ modern computer languages, environments, and platforms in creating innovative career paths to be an entrepreneur, and a zest for higher studies and research.

Dr. Babasaheb Ambedkar Technological University

Syllabus & Scheme

Course Code	Course Title	Weekly Teaching hrs.	Evaluation Scheme			Credit
			MSE	CA	ESE	
BTCOS407	Web Technology Lab (Seminar-II)	**4**	**-**	**60**	**40**	**2**

BTCOS407 Web Technology (Seminar-II) Laboratory

Shri Vile Parle Kelavani Mandal’s

Institute of Technology, Dhule.

www.svkm-iot.ac.in

Survey No. 499, Plot No.2, Behind Gurudwara, Mumbai-Agra Road, Dhule(M.S.)

DEPARMENT OF COMPUTER ENGG.

Academic Year: 2022-23

LABORATORY : BTCOS407 Web Technology Lab (Seminar-II)	**SEMESTER : 4th Sem**

Master List of Practical

SR. NO.	PRACTICAL DESCRIPTION
1	Design an html form for displaying information using interactive CSS including images, tables.
2	Design a webpage describing department with following specification: • Insert an image and create a link such that clicking on image takes user to other page. • Also apply font styling like italics, underline and two other fonts to words you find appropriate also use header tags.
3	Write a JavaScript to design a simple calculator to perform the following operations: sum, product, difference and quotient.
4	Write a JavaScript to validate the following fields of employee on html form: email, name, mobile no., address, salary.
5	Develop and demonstrate a HTML file that includes JavaScript script that uses functions for the following problems: • Parameter: A string Output: Length of the String • Parameter: A number Output: The number with its digits in the reverse order
6	Develop and demonstrate a HTML file that includes JavaScript for the following problems: • Input: A starting and ending number • b. Output: find all the prime numbers between starting and ending number.
7	Write a PHP program to display a digital clock which displays the current time of the server.
8	Write a PHP program to implement sign-In and Sign-out functionality
9	Write a PHP program to keep track of the number of visitors visiting the Web page and to display this count of visitors, with proper headings.
10	Write a PHP code to implement AJAX functionality
11	Write a PHP program to perform search operation on the student records using AJAX.

12	Write a PHP program to sort the student records which are stored in the database using ascending/descending order.

Subject In-charge: Prof. Amirkhan Pinjari /Prof Rinku Sharma Department of Computer Engineering	**APPROVED BY:** Dr. Makarand Shahade HOD, Department of Computer Engineering

Rubrics for Assessment

Assignment/Experiment :

Date of Performance: **Date of submission :**

Marks Split Up to	Maximum Marks	Marks Obtained
Performance/ conduction	3	
Report Writing	3	
Attendance	2	
Viva/Oral	2	
Total Marks	10	
Signature of Subject Teacher		

Shri Vile Parle Kelavani Mandal's

INSTITUTE OF TECHNOLOGY

DHULE (M.S.)

DEPARMENT OF COMPUTER ENGINEERING

Subject : Web Technology Lab **Subject Code :** BTCOS407

Class: S. Y. Comp. Engg. **Expt. No. :** 01

Title : Design an html form for displaying information using interactive CSS including images, tables.

Problem Staement: The main aim of this lab is to demonstrate HTML Tags and design webpage using HTML Tags

Software Required : VS Code , Chrome Browser

Theory :

HTML stands for Hyper Text Markup Language.HTML is the standard markup language for creating Web pages.HTML describes the structure of a Web page.HTML consists of a series of elements.HTML elements tell the browser how to display the content.HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

An HTML element is defined by a start tag, some content, and an end tag:

<tagname> Content goes here... </tagname>

The HTML element is everything from the start tag to the end tag:

<h1>My First Heading</h1>

<p>My first paragraph. </p>

All HTML documents must start with a document type declaration: <!DOCTYPE html>.The HTML document itself begins with <html> and ends with </html>.The visible part of the HTML document is between <body> and </body>.he <!DOCTYPE> declaration represents the document type, and helps browsers to display web pages correctly.It must only appear once, at the top of the page (before any HTML tags).The <!DOCTYPE> declaration is not case sensitive.The <!DOCTYPE> declaration for HTML5 is:

Tags:

HTML paragraphs are defined with the <p> tag:

HTML links are defined with the <a> tag:

<a href="https://www.w3schools.com">This is a link</a>

The images are defined with the <img> tag.

The source file (src), alternative text (alt), width, and height are provided as attributes:

Example <img src="w3schools.jpg" alt="W3Schools.com" width="104" height="142">

Conclusion :

1. The HTML is used to design the structure of webpage.
2. HTML tags are used to implement different HTML elements according to their functionalities.
3. Registration form can be created using HTML form tags and form elements.

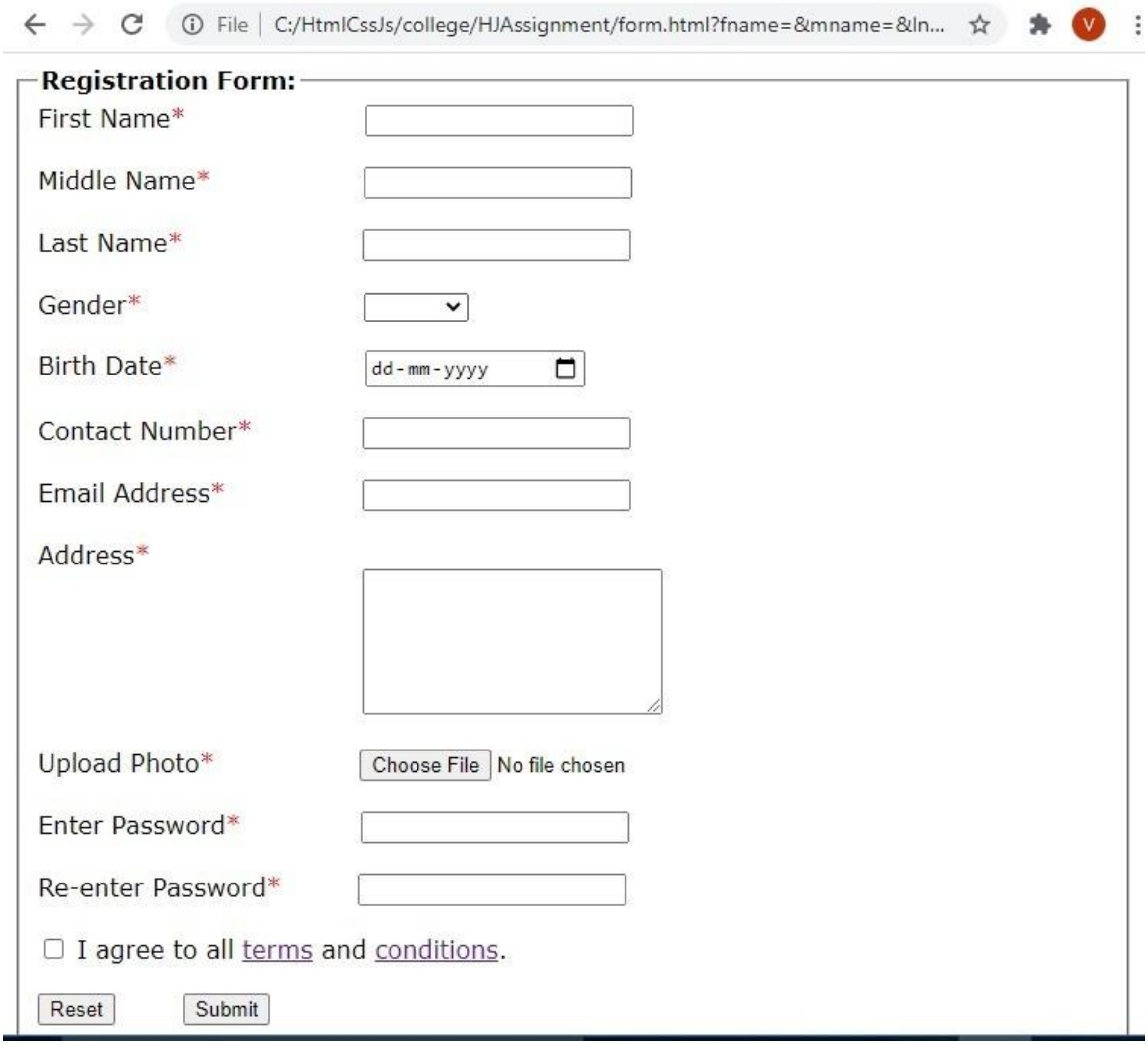
Code:

```
 <!DOCTYPE html>

 <html>
     <head>
     <title>
         Registration form using HTML
     </title>
 </head>
 <body>
     <form style="font-family: Verdana, Geneva, Tahoma, sans-serif;
 background-color;">
         <fieldset>
             <legend>
                <b>Registration Form:</b>
             </legend>
               <label for="fname">First Name<span style="color: red
     ;">*</span>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp
 ;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nb
 sp;</label>

<input type="text" id="fname" name="fname"><br><br>

 <label for="mname">Middle Name<span style="color: re
 d;">*</span>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbs
 p;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
 </label>
<input type="text" id="mname" name="mname"><br><br>
```

```
            <label for="lname">Last Name<span style="color: red;
">*</span>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbs
p;</label>
            <input type="text" id="lname" name="lname"><br><br>
            <label for="gender">Gender<span style="color: red;">
*</span>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&n
bsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</label>
            <select id="gender" name="gender">

                <option value="Select"></option>
                <option value="Male">Male</option>
                <option value="Female">Female</option>
                <option value="Other">Other</option>
            </select><br><br>

              <label for="birthDate">Birth Date<span style="color:
    red;">*</span>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&
   nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp
;&nbsp;&nbsp;</label>

<input type="date" id="birtDate" name="birtDate"><br><br>
  <label for="email">Contact Number<span style="color:
  red;">*</span>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&
nbsp;&nbsp;&nbsp;&nbsp;</label>

            <input type="text" id="email" name="email"><br><br>

            <label for="email">Email Address<span style="color:
red;">*</span>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&n
bsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</label>

<input type="text" id="email" name="email"><br><br>
            <label for="address">Address<span style="color: red;
">*</span></label><br>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
            &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbs
p;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&n
bsp;&nbsp;&nbsp;&nbsp;
            <textarea id="address" name="address" rows="6" cols=
"25"></textarea><br><br>


<label for="photo">Upload Photo<span style="color: r
ed;">*</span>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nb
sp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</label>
            <input type="file" id="photo" name="photo"><br><br>

<label for="pass">Enter Password<span style="color:
red;">*</span>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&n
bsp;&nbsp;&nbsp;&nbsp;&nbsp;</label>
 <input type="password" id="pass" name="pass"><br><br>

<label for="pass">Re-enter Password<span style="color:
red;">*</span>&nbsp;&nbsp;&nbs
p;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</label>
<input type="password" id="rpass" name="rpass"><br><br>
<input type="checkbox" id="agreement" name="agreement">
```

```
             I agree to all <a href="" target="_blank">terms</a>
and <a href="" target="_blank">conditions</a>.<br>
             <br>
<input  type="reset"
value="Reset">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
             <input type="submit" value="Submit">

         </fieldset>

    </form>

</body>

</html>
```

OUTPUT



File | C:/HtmlCssJs/college/HJAssignment/form.html?fname=&mname=&ln...

Registration Form:

First Name*

Middle Name*

Last Name*

Gender*

Birth Date* dd-mm-yyyy

Contact Number*

Email Address*

Address*

Upload Photo* Choose File No file chosen

Enter Password*

Re-enter Password*

☐ I agree to all terms and conditions.

Reset Submit

Shri Vile Parle Kelavani Mandal's

INSTITUTE OF TECHNOLOGY

DHULE (M.S.)

DEPARMENT OF COMPUTER ENGINEERING

Subject : Web Technology Lab **Subject Code :** BTCOS407

Class: S. Y. Comp. Engg. **Expt. No. :** 02

Title : Design a webpage describing department with Image Tag and Anchor Tag and also applying Font Styling with header Tags.

Problem Staement :

Create a webpage with HTML describing your department with following specification:

- Change the background color of the page. At the bottom create a link to take user to the top of the page.
- Insert an image and create a link such that clicking on image takes user to other page.
- Also apply font styling like italics, underline and two other fonts to words you find appropriate. Also use header tags.

Software Required : VS Code , Chrome Browser

Theory :

The HTML <img> tag is used to embed an image in a web page.Images are not technically inserted into a web page; images are linked to web pages. The <img> tag creates a holding space for the referenced image.The <img> tag is empty, it contains attributes only, and does not have a closing tag.The <img> tag has two required attributes:

src - Specifies the path to the image
alt - Specifies an alternate text for the image

The src Attribute

The required src attribute specifies the path (URL) to the image.

Note: When a web page loads, it is the browser, at that moment, that gets the image from a web server and inserts it into the page. Therefore, make sure that the image actually stays in the same spot in relation to the web page, otherwise your visitors will get a broken link icon. The broken link icon and the alt text are shown if the browser cannot find the image.

Example

```
<img src="img_chania.jpg" alt="Flowers in Chania">
```

HTML Headings:

HTML headings are defined with the <h1> to <h6> tags.
<h1> defines the most important heading. <h6> defines the least important heading.

Example

```
<h1>Heading 1</h1>
<h2>Heading 2</h2>
<h3>Heading 3</h3>
<h4>Heading 4</h4>
<h5>Heading 5</h5>
<h6>Heading 6</h6>
```

HTML Text Formatting

Formatting elements were designed to display special types of text:

- <b> - Bold text
- <strong> - Important text
- <i> - Italic text
- <em> - Emphasized text
- <mark> - Marked text
- <small> - Smaller text
- <del> - Deleted text
- <ins> - Inserted text
- <sub> - Subscript text
- <sup> - Superscript text

Conclusion:

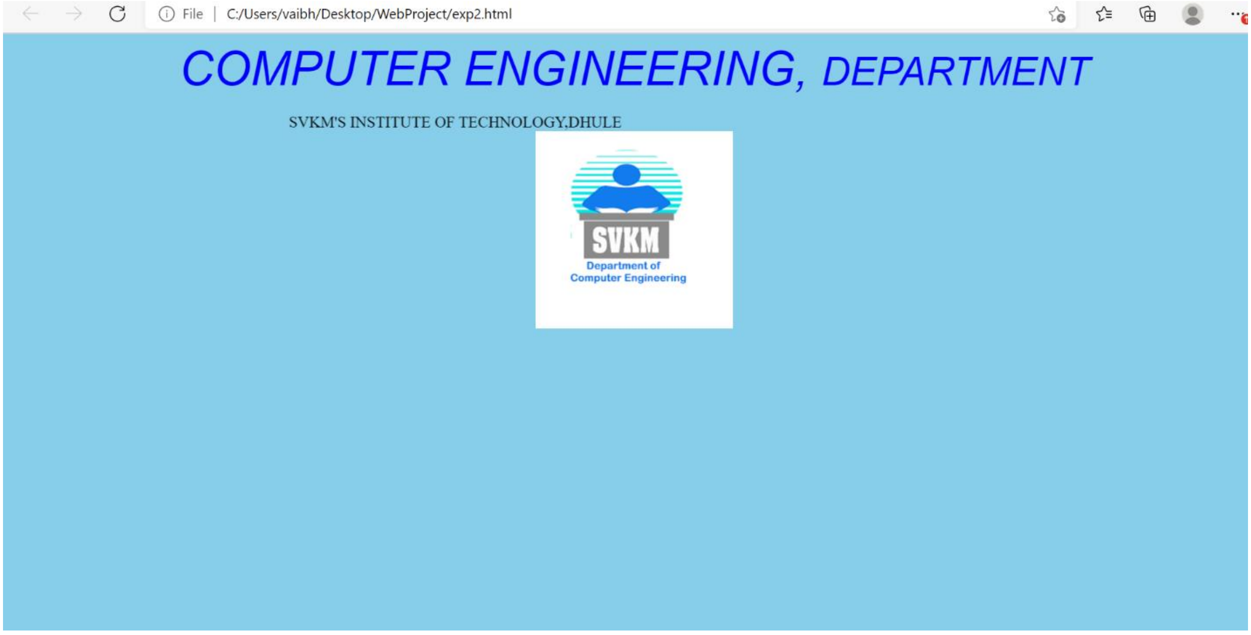
1. The HTML is used to design the structure of webpage.
2. HTML tags are used to implement different HTML elements according to their functionalities.

CODE

HTML Program

```
<Html>
<head><title><center><u>my department</u> </center></title></head>
<body bgcolor="skyblue">
<font size="7" face="arial" color="blue"><center><i>computer engineering,
<small>department<center></small></i></font><br>
<body bgcolor="pink"<center>
<marquee behaviour="slide"> svkm's institute of technology, dhule </marquee>
<a href="https://www.svkm-iot.ac.in/computer-engineering/"><img src="svkmcomp.png"
width="200" height="200"></img></a>
</body>
</html>
```

Output



COMPUTER ENGINEERING, DEPARTMENT

SVKM'S INSTITUTE OF TECHNOLOGY,DHULE

Shri Vile Parle Kelavani Mandal's

INSTITUTE OF TECHNOLOGY

DHULE (M.S.)

DEPARMENT OF COMPUTER ENGINEERING

Subject : Web Technology Lab **Subject Code :** BTCOS407

Class: S. Y. Comp. Engg. **Expt. No. :** 03

Title : Write a JavaScript to design a simple calculator.

Problem Staement : Write a JavaScript to design a simple calculator to perform the following operations:

1. **Sum,**
2. **Product**
3. **Difference**
4. **Quotient**

Software Required : VS Code , Chrome Browser

Theory :

Basic HTML tags:
The following tags are used while creating the web based calculator.

- <script>
  The <script> tag is used to embed Javascript inside html document to add functionalityto web page.
- <link>
  The <link> tag is used to embed external styles sheet or documents to the current HTML document.
- <h1>
  The <h1> defines the most important heading
- <div>
  The <div> tag defines a division or a section in an HTML document.

Form tags:

- <button>
  The <button> tag defines a clickable button.
- <input>
  The <input> tag specifies an input field where the user can enter data.

JavaScript:

JavaScript, often abbreviated as JS, is a programming language that conforms to the ECMA Script specification. JavaScript is high-level, often just-in-time compiled, and multi-paradigm. It has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-classfunctions.

- JavaScript statements are composed of Values, Operators, Expressions, Keywords, and Comments.
- Most JavaScript programs contain many JavaScript statements. The statements are executed, one by one, in the same order as they are written.
- Semicolons separate JavaScript statements.
- JavaScript statements can be grouped together in code blocks, inside curly brackets {...}. The purpose of code blocks is to define statements to be executed together.

Variable Declaration in Javascript:

In a programming language, variables are used to store data values. JavaScript uses the var keyword to declare variables. An equal sign (=) is used to assign values to variables.For e.g.

```
var x;

        x = 10;
        myName = "Ajay";y = x + 20;
```

Javascript Operators:

- JavaScript uses arithmetic operators (+, -, *, /) to compute values.
- JavaScript uses an assignment operator (=) to assign values to variables.

A. Arithmetic operators: There are eight arithmetic operators in Javascript.

- Addition (+)
- Subtraction (-)
- Multiplication (*)
- Division (/)
- Exponential (**)
- Modulus (%)
- Increment (++)
- Decrement (--)

B. Assignment operators: Assignment operators assign value to the variable.

Assignment operators are =, - =, + =, * =, / =, % =, ** =.For e.g. var
x = 5;
X += 5;

C. Logical Operators: Logical operators are used to perform logical operation such as and,or and not.
   - Logical And (&&)
   - Logical Or (||)
   - Logical Not (!)

D. Type Operators:
   - typeof: Returns the type of variable.
   - instanceof: Returns true if an object is an instance of an object type.

E. Bitwise Operators:
   - And (&)
   - Or (|)
   - Not (~)
   - Xor (^)
   - Zero fill left shift (<<)
   - Zero fill right shift(>>>)
   - Signed right shift(>>)Javascript

Functions:

A JavaScript function is a block of code designed to perform a particular task. A JavaScriptfunction is executed when "something" invokes it (calls it).

For e.g.

```
function myFunction(p1, p2) {
return p1 + p2;              // The function returns the addition of p1
//and p2
}
```

Syntax of Javascript function:

- A JavaScript function is defined with the ‘function’ keyword, followed by a name, followed by parentheses ().

- Function names can contain letters, digits, underscores, and dollar signs (same rules as variables).

- The parentheses may include parameter names separated by commas: (parameter1, parameter2, ...)

- The code to be executed, by the function, is placed inside curly brackets ‘{}’.

  For e.g.
  ```
  function name(parameter1, parameter2, parameter3) {
    // code to be executed
  }
  ```

- Function parameters are listed inside the parentheses () in the function definition.

- Function arguments are the values received by the function when it is invoked.

- Inside the function, the arguments (the parameters) behave as local variables.

Function Calling:

The code inside the function will execute when "something" invokes (calls) the function:

- When an event occurs (when a user clicks a button)
- When it is called from JavaScript code
- Automatically (self calling i.e. recursion)Function

Return:

When JavaScript reaches a ‘return’ statement, the function will stop executing.

If the function was invoked from a statement, JavaScript will "return" to execute the code afterthe invoking statement.

Functions often compute a return value. The return value is "returned" back to the "caller".Example of

function is given below:

```
var x = addition(4, 3);                    // Function is called, return valuewill //end up in x

function addition(a, b) {
    return a + b;                          // Function returns the addition ofa //and b
}
```

Conclusion:

1. JavaScript is use to add functionality to the HTML document.
2. HTML tags can be use to make web page interactive.
3. Form tags and can be use to make webpage dynamic.
4. Javascript supports variety of operators and operation.
5. Javascript functions can be use for different operations and can be make web page more interactive.

Code:

HTML:

```
<!DOCTYPE html>
<html>
<head>
<title>Web Based Calculator</title>
<link rel="stylesheet" href="calc.css">
</head>
<body>
<h1>Calulator with HTML, CSS and JAVASCRIPT</h1>
<script src="calc.js"></script>
<div id="calcBody"class="container">
<input id="id1" type="text">
<div id="outputField"class="ioField">
<p id="output"></p>
</div>

<div id="inputArea"class="ioField">
<div id ="on"class="input" onclick="onFunction()">
<button id="onbtn" class="inpbtn">ON</button>
</div>

<div id = "off" class="input" onclick="printName()">
<button id="offbtn" class="inpbtn">OFF</button>
</div>


<div id ="del"css="input" onclick="delBack()">
<button id="delbtn" class="inpbtn">DEL</button>
</div>

<div id ="ac"class="input" onclick="clearAll()">
<button id="acbtn" class="inpbtn">AC</button>
</div>

<div class="input">
<button id="inp1" class="inpbtn" onclick="myInput1()">1</button>
</div>


<div class="input">
<button id="inp2" class="inpbtn" onclick="myInput2()"> 2 </button>
</div>


<div class="input">
<button id="inp3" class="inpbtn" onclick="myInput3()">3 </button>
</div>

<div class="input">
<button id="inp4" class="inpbtn" onclick="myInput4()"> 4 </button>
```

```
</div>

<div class="input">
<button id="inp5" class="inpbtn" onclick="myInput5()">5 </button>
</div>

<div class="input">
<button id="inp6" class="inpbtn" onclick="myInput6()">6 </button>
</div>

<div class="input">
<button id="inp7" class="inpbtn" onclick="myInput7()"> 7 </button>
</div>

<div class="input">
<button id="inp8" class="inpbtn" onclick="myInput8()">8</button>
</div>

<div class="input">
<button id="inp9" class="inpbtn" onclick="myInput9()">9 </button>
 </div>

<div class="input">
<button id="inp0" class="inpbtn" onclick="myInput0()">0</button>
</div>

<div id="decimal"class="input">
<button id="inpdml" class="inpbtn" onclick="myInputDec()">.</button>
</div>

<div id="addition"class="input">
<button id="inpplus" class="inpbtn" onclick="myInputAdd()">+</button>
</div>

<div id="substraction"class="input">
<button id="inpminus" class="inpbtn" onclick="myInputSub()">-</button>
</div>

<div id="multiplicatio"class="input">
<button id="inpmult" class="inpbtn" onclick="myInputMult()">x</button>
</div>

<div id="division"class="input">
<button id="inpdiv" class="inpbtn" onclick="myInputDiv()"> / </button>
</div>

<div id="equal" class="input" onclick="calAnswer()">
<button id="inpeql" class="inpbtn">=</button>
</div>


</div>
</div>
```

```
</body>
</html>
```

CSS:

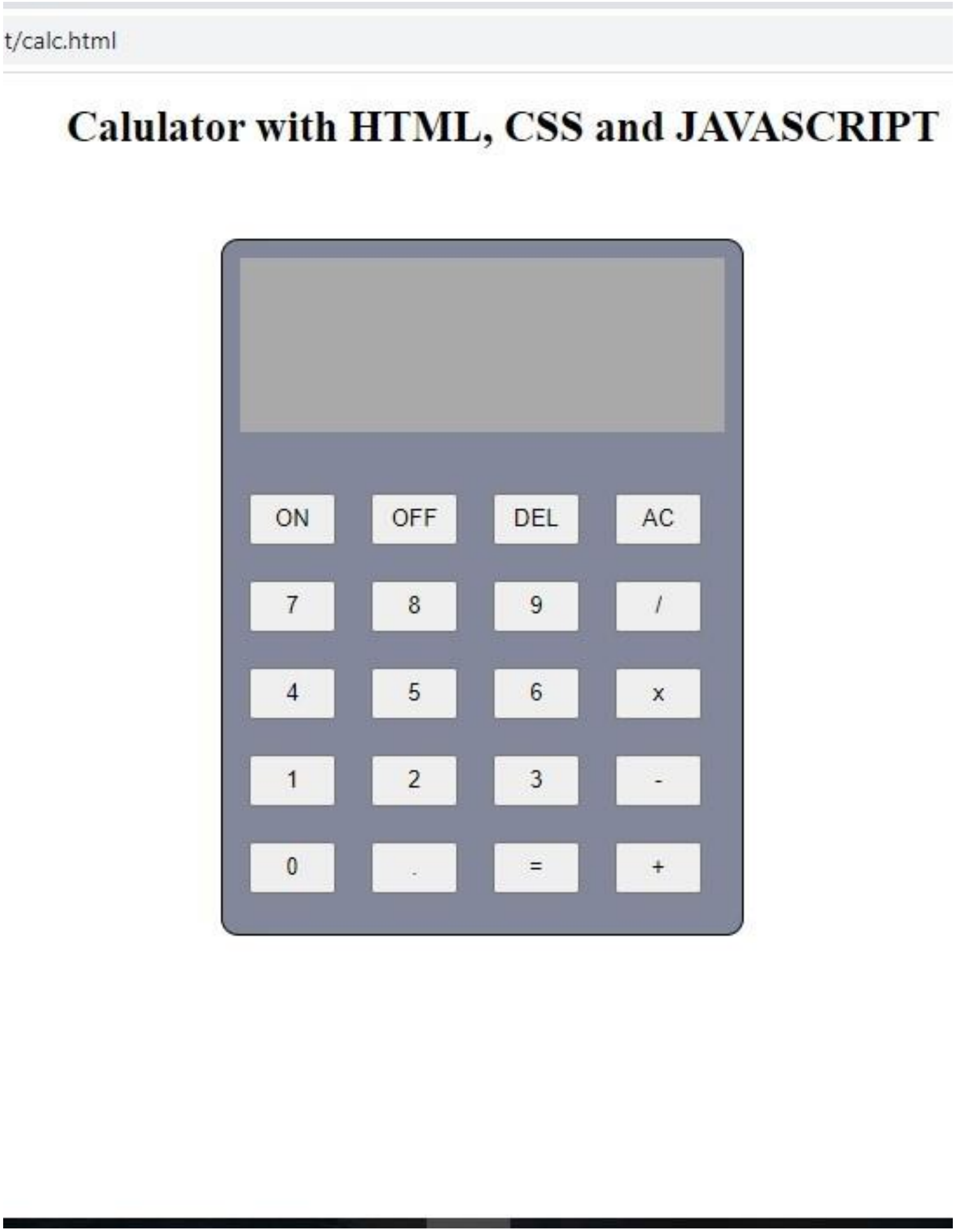
```
h1{
    font-family: 'Times New Roman', Times, serif;

    font-size: 25px;

    text-align: center;
}

#calcBody{

    width: 300px;

    height: 400px;

    box-sizing: border-box;

    border: 1px black solid;

    background: rgb(130, 135, 156);

    margin-top: 50px;

    margin-left: 38%;

    padding: 5px;

    border-radius: 10px;

}

#id1{
    margin : 5px;

    margin-bottom:0;

    background-color: darkgray;

    height: 40px;

    width: 278px;

    border: none;

    text-align: left;

    font-size: 24px;
    font: bold;
    font-family: 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open Sans', 'Helvetica Neue', sans-serif;
    padding: 10px;
    box-sizing: border-box;
}
#outputField{
    margin : 5px;
    margin-top:0;
    background-color: darkgray;
    text-align: end;
    font-size: 30px;
    font:italic;
    font-family: Verdana, Geneva, Tahoma, sans-serif;
    height: 60px;
    padding: 10px;
    box-sizing: border-box;
}
#output{
    margin: 0px;
    font: italic;
}
.input{
    width  :  50px;
    height:   30px;
    float: left;
    margin: 10px;

}
#inputArea{
    margin-top: 25px;
}
.inpbtn{
    width: 50px;
    height: 30px;
}
```

JavaScript Code:

```
function calAnswer(){
    var exp = document.getElementById("id1").value;var
    res = eval(exp);
    document.getElementById("output").innerHTML= res+'.';
}
function clearAll(){ document.getElementById("id1").value
    = ""; document.getElementById("output").innerHTML =
    '0.';
}
function onFunction(){
    document.getElementById("output").innerHTML = '0.'
}

function delBack(){
    ex = document.getElementById("id1").value;
    document.getElementById("id1").value = ex.slice( 0, ex.length -
 1);
}
function   printName(){ document.getElementById("id1").value
    = ""; document.getElementById("output").innerHTML =
    "MHITS";setTimeout(offCalc, 1000);
```

```
}

function offCalc(){

    document.getElementById("id1").value = "";

    document.getElementById("output").innerHTML= "";

}

function myInput1(){

    document.getElementById("id1").value += '1';

}

function myInput2(){

    document.getElementById("id1").value += '2';

}

function myInput3(){

    document.getElementById("id1").value += '3';

}

function myInput4(){

    document.getElementById("id1").value += '4';

}

function myInput5(){

    document.getElementById("id1").value += '5';

}

function myInput6(){

    document.getElementById("id1").value += '6';

}

function myInput7(){

    document.getElementById("id1").value += '7';

}

function myInput8(){

    document.getElementById("id1").value += '8';

}

function  myInput9(){
    document.getElementById("id1").value += '9';
}
function myInput0(){
    document.getElementById("id1").value += '0';
}
function myInputDec(){
    document.getElementById("id1").value += '.';
}
function myInputAdd(){
    document.getElementById("id1").value += '+';
}
function myInputSub(){
    document.getElementById("id1").value += '-';
}
function myInputMult(){
    document.getElementById("id1").value += '*';
}
function myInputDiv(){
    document.getElementById("id1").value += '/';
}
```

Output



t/calc.html

Calulator with HTML, CSS and JAVASCRIPT

Shri Vile Parle Kelavani Mandal's

INSTITUTE OF TECHNOLOGY

DHULE (M.S.)

DEPARMENT OF COMPUTER ENGINEERING

Subject : Web Technology Lab **Subject Code :** BTCOS407

Class: S. Y. Comp. Engg. **Expt. No. :** 04

Title : Write a JavaScript to validate the following fields of employee on html form: email, name, mobile no., address, salary.

Problem Statement : Write a JavaScript to validate the following fields of employee on html form: email, name, mobile no., address, salary

Software Required : V S Code ,Chrome Browser

Theory:

It is important to validate the form submitted by the user because it can have inappropriate values. So, validation is must to authenticate user. JavaScript provides facility to validate the form on the client-side so data processing will be faster than server-side validation. Most of the web developers prefer JavaScript form validation. Through JavaScript, we can validate name, password, email, date, mobile numbers and more fields.

Conclusion:

- In this article, we learned about form verification using HTML, CSS, and JavaScript.
- A form is also called a web form or HTML form.
- Forms are used on web pages for users to enter their required details, which are sent to the server for processing.

Code

HTML program

```
<!DOCTYPE html>
<html>
<head>
    <title>Reg Form</title>
```

```
</head>
<body>
    <center><h1>Form Validation using HTML,CSS,JavaScript</h1></center>
    <hr>
    <form method="" action="" name="reg_form" onsubmit="return validate()">
        <h2>Registration Form</h2>
        <table>
            <tr>
                <td><label>First Name: </label></td>
                <td>
                    <input type="text" name="fname" placeholder="First Name">
                </td>
            </tr>
            <tr>
                <td><label>Last Name: </label></td>
                <td>
                    <input type="text" name="lname" placeholder="Last Name">
                </td>
            </tr>
            <tr>
                <td><label>Address: </label></td>
                <td>
                    <input type="textarea" size="50" name="address" placeholder="Address">
                </td>
            </tr>
            <tr>
                <td><label>Gender: </label></td>
                <td>
                    <input type="radio" name="gender" value="male">Male
                    <input type="radio" name="gender" value="femele">Female
                </td>
            </tr>
            <tr>
                <td><label>Email Id: </label></td>
        <td>
          <input type="text" name="email" placeholder="example@gmail.com">
        </td>
      </tr>
      <tr>
        <td><label>Mobile: </label></td>
        <td>
          <input type="number" name="mobile">
        </td>
      </tr>
      <tr>
      <td><label>Course: </label></td>
        <td>
          <select name="course">
            <option value="select course">select course</option>
            <option value="HTML">HTML</option>
            <option value="CSS">CSS</option>
            <option value="JavaScript">JAVASCRIPT</option>
            <option value="Java">JAVA</option>
          </select>
        </td>
      </tr>
      <tr>
        <td>
          <input type="submit" name="submit" value="Submit">
          <input type="reset" name="reset" value="Reset">
        </td>
      </tr>
    </table>
  </form>
</body>
</html>
```

CSS Code

```
<style type="text/css">
    body{
        font-family: Calibri;
    }
    input[type="text"] {
        width: 250px;
    }
    input[type="submit"], input[type="reset"] {
        width: 77px;
        height: 27px;
        position: relative;left: 180px;
    }
    form{
        text-align: center;
        font-family: Calibri;
        font-size: 20px;
        border: 3px solid black;
        width: 600px;
        margin: 20px auto;
    }
    td {
        padding: 10px;
    }
    td:first-child {
        text-align: right;
        font-weight: bold;
    }
    td:last-child {
        text-align: left;
</style>
```

JavaScript

```
<script>
    function validate() {
        var fname = document.reg_form.fname;
        var lname = document.reg_form.lname;
        if (fname.value.length <= 0) {
            alert("Name is required");
            fname.focus();
            return false;
        }
        if (lname.value.length <= 0) {
            alert("Last Name is required");
            lname.focus();
            return false;
        }
</script>

<script>
var address = document.reg_form.address;
var gender = document.reg_form.gender;
if (address.value.length <= 0) {
    alert("Address is required");
    address.focus();
    return false;
    }
if (gender.value.length <= 0) {
    alert("Gender is required");
    gender.focus();
    return false;
  }
</script>
```

Here is my JavaScript code for Form validations this.

```
<!DOCTYPE html>
<html>
```

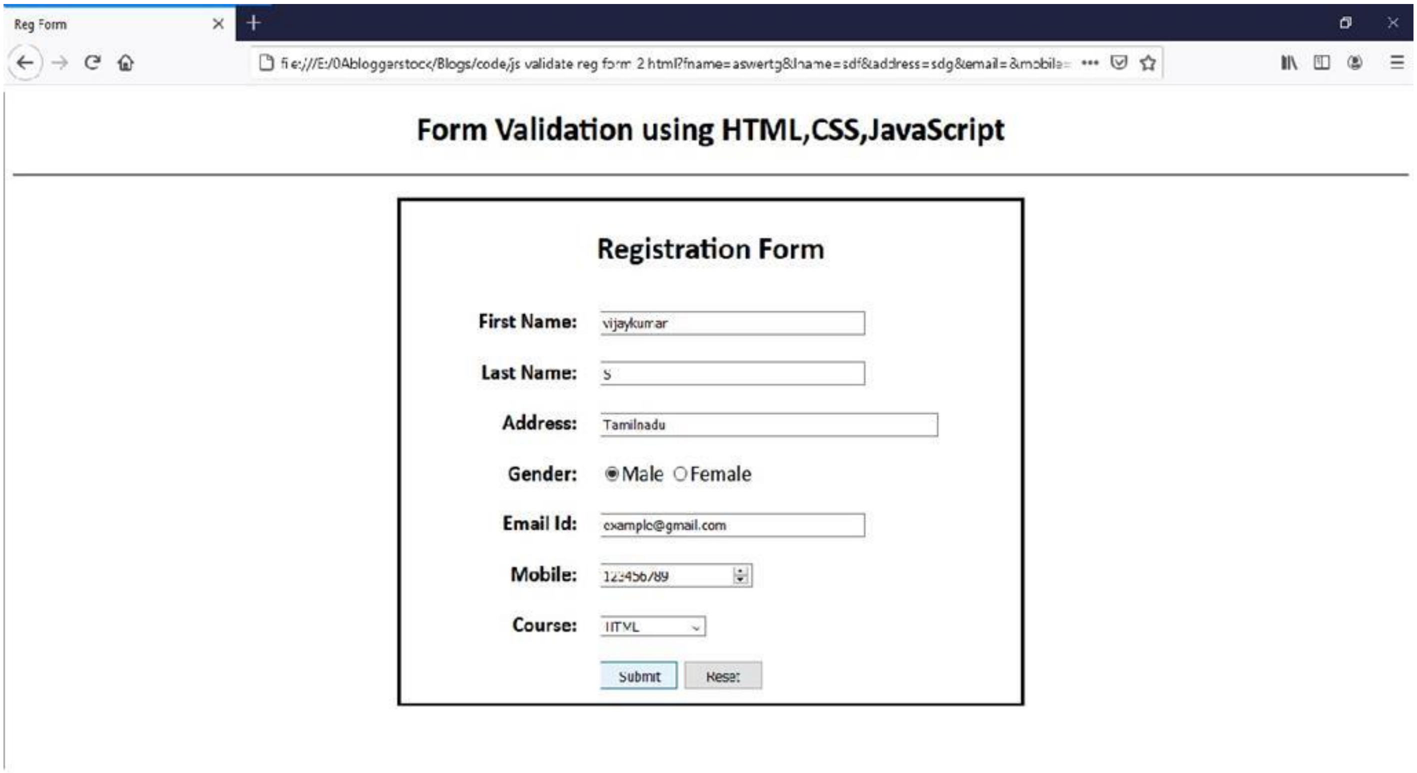
```
<head>
    <title>Reg Form</title>
    <style type="text/css">
        body{
            font-family: Calibri;
        }
        input[type="text"] {
            width: 250px;
        }
        input[type="submit"], input[type="reset"] {
            width: 77px;
            height: 27px;
            position: relative;left: 180px;
        }
        form{
            text-align: center;
            font-family: Calibri;
            font-size: 20px;
            border: 3px solid black;
            width: 600px;
            margin: 20px auto;
        }
        td {
            padding: 10px;
        }
        td:first-child {
            text-align: right;
            font-weight: bold;
        }
        td:last-child {
            text-align: left;

    </style>
    <script>
```

```
function validate() {
    var fname = document.reg_form.fname;
    var lname = document.reg_form.lname;
    var address = document.reg_form.address;
    var gender = document.reg_form.gender;
    var email = document.reg_form.email;
    var mobile = document.reg_form.mobile;
    var course = document.reg_form.course;

    if (fname.value.length <= 0) {
        alert("Name is required");
        fname.focus();
        return false;
    }
    if (lname.value.length <= 0) {
        alert("Last Name is required");
        lname.focus();
        return false;
    }
    if (address.value.length <= 0) {
        alert("Address is required");
        address.focus();
        return false;
    }
    if (gender.value.length <= 0) {
        alert("Gender is required");
        gender.focus();
        return false;
    }
    if (email.value.length <= 0) {
        alert("Email Id is required");
        email.focus();
        return false;
    }
```

```
            if (mobile.value.length <= 0) {
                alert("Mobile number is required");
                mobile.focus();
                return false;
            }
            if (course.value == "select course") {
                alert("Course is required");
                course.focus();
                return false;
            }
            return false;
        }
    </script>
</head>
<body>
    <center><h1>Form Validation using HTML,CSS,JavaScript</h1></center>
    <hr>
    <form method="" action="" name="reg_form" onsubmit="return validate()">
        <h2>Registration Form</h2>
        <table>
            <tr>
                <td><label>First Name: </label></td>
                <td>
                    <input type="text" name="fname" placeholder="First Name">
                </td>
            </tr>
            <tr>
                <td><label>Last Name: </label></td>
                <td>
                    <input type="text" name="lname" placeholder="Last Name">
                </td>
            </tr>
            <tr>
                <td><label>Address: </label></td>
                <td>
                    <input type="textarea" size="50" name="address" placeholder="Address">
                </td>
            </tr>
            <tr>
                <td><label>Gender: </label></td>
                <td>
                    <input type="radio" name="gender" value="male">Male
                    <input type="radio" name="gender" value="femele">Female
                </td>
            </tr>
            <tr>
                <td><label>Email Id: </label></td>
                <td>
                    <input type="text" name="email" placeholder="example@gmail.com">
                </td>
            </tr>
            <tr>
                <td><label>Mobile: </label></td>
                <td>
                    <input type="number" name="mobile">
                </td>
            </tr>
            <tr>
            <td><label>Course: </label></td>
                <td>
                    <select name="course">
                        <option value="select course">select course</option>
                        <option value="HTML">HTML</option>
                        <option value="CSS">CSS</option>
                        <option value="JavaScript">JAVASCRIPT</option>
                        <option value="Java">JAVA</option>
                    </select>
                </td>
```

```
                        </tr>
                        <tr>
                            <td>
                                <input type="submit" name="submit" value="Submit">
                                <input type="reset" name="reset" value="Reset">
                            </td>
                        </tr>
                    </table>
                </form>
            </body>
</html>
```

Output

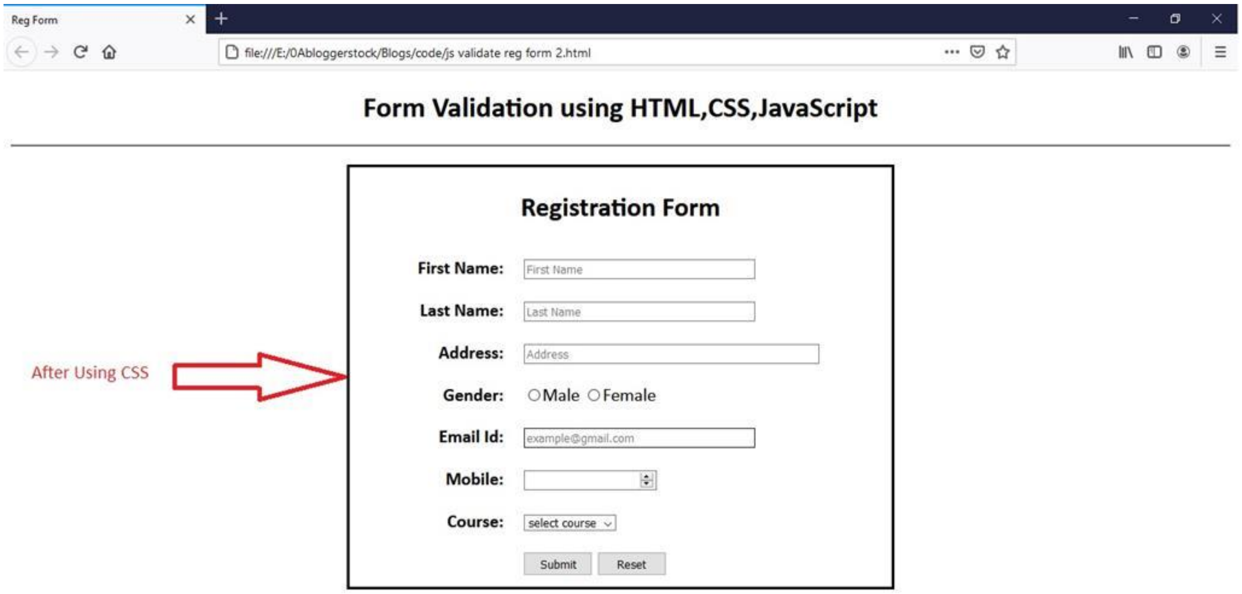


file:///E:/0Abloggerstock/Blogs/code/js validate reg form 2.html?fname=aswertg&lname=sdf&address=sdg&email=&mobile=

Form Validation using HTML,CSS,JavaScript

Registration Form

First Name:	vijaykumar
Last Name:	S
Address:	Tamilnadu
Gender:	● Male ○ Female
Email Id:	example@gmail.com
Mobile:	123456789
Course:	HTML



file:///E:/0Abloggerstock/Blogs/code/js validate reg form 2.html

Form Validation using HTML,CSS,JavaScript

Registration Form

First Name:	First Name
Last Name:	Last Name
Address:	Address
Gender:	○ Male ○ Female
Email Id:	example@gmail.com
Mobile:	
Course:	select course

After Using CSS

Shri Vile Parle Kelavani Mandal's

INSTITUTE OF TECHNOLOGY

DHULE (M.S.)

DEPARMENT OF COMPUTER ENGINEERING

Subject : Web Technology Lab **Subject Code :** BTCOS407

Class: S. Y. Comp. Engg. **Expt. No. :** 05

Title : Develop and demonstrate a HTML file that includes JavaScript that uses string functions and Reverse No.

Problem Staement : Design a HTML file that includes JavaScript that uses string functions and Reverse No.

Software Required : V S Code , Crome Browser

Theory :

Methods	Description
charAt()	It provides the char value present at the specified index.
charCodeAt()	It provides the Unicode value of a character present at the specified index.
concat()	It provides a combination of two or more strings.
indexOf()	It provides the position of a char value present in the given string.
lastIndexOf()	It provides the position of a char value present in the given string by searching a character from the last position.
search()	It searches a specified regular expression in a given string and returns its position if a match occurs.
match()	It searches a specified regular expression in a given string and returns that regular expression if a match occurs.
replace()	It replaces a given string with the specified replacement.
substr()	It is used to fetch the part of the given string on the basis of the specified starting position and length.

substring()	It is used to fetch the part of the given string on the basis of the specified index.
slice()	It is used to fetch the part of the given string. It allows us to assign positive as well negative index.
toLowerCase()	It converts the given string into lowercase letter.
toLocaleLowerCase()	It converts the given string into lowercase letter on the basis of host?s current locale.
toUpperCase()	It converts the given string into uppercase letter.
toLocaleUpperCase()	It converts the given string into uppercase letter on the basis of host?s current locale.
toString()	It provides a string representing the particular object.
valueOf()	It provides the primitive value of string object.
split()	It splits a string into substring array, then returns that newly created array.
trim()	It trims the white space from the left and right side of the string.

Conclusion:

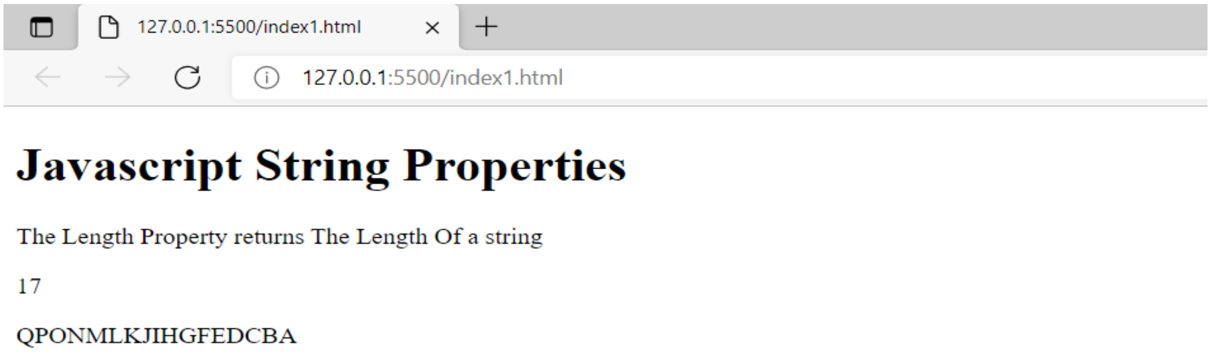
- In this practical, we learned about various function of JavaScript.

CODE:

1.Parameter:A String

```
<!DOCTYPE html>
<html lang="en">
<body>

    <h1>Javascript String Properties</h1>
       <p>The Length Property returns The Length Of a string</p>
   <p id="demo"></p>
   <script>
        function ReverseString(str)
        {
            return str.split('').reverse().join('')
        }
        document.write(ReverseString("ABCDEFGHIJKLMOPQ"));
        let text="ABCDEFGHIJKLMNOPQ";
        document.getElementById("demo").innerHTML=text.length;
    </script>

</body>
</html>
```

output



Javascript String Properties

The Length Property returns The Length Of a string

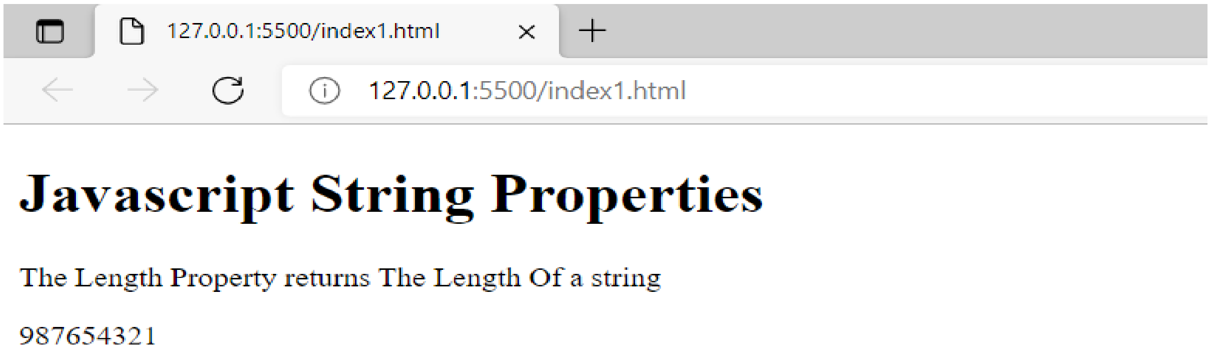
17

QPONMLKJIHGFEDCBA

2.Parameter: A Number

```
<!DOCTYPE html>
<html lang="en">
<body>
        <h1>Javascript String Properties</h1>
        <p>The Length Property returns The Length Of a tring</p>
   <p id="demo"></p>
   <script>
        function ReverseString(str)
        {
            return str.split('').reverse().join('')
        }
        document.write(ReverseString("123456789"));

    </script>
```

```
</body>
</html>
```



Javascript String Properties

The Length Property returns The Length Of a string

```
987654321
```

Shri Vile Parle Kelavani Mandal's

INSTITUTE OF TECHNOLOGY

DHULE (M.S.)

DEPARMENT OF COMPUTER ENGINEERING

Subject : Web Technology Lab **Subject Code :** BTCOS407

Class: S. Y. Comp. Engg. **Expt. No. :** 06

Title : Develop and demonstrate a HTML file that includes JavaScript for finding prime numbers in given range.

Problem Staement : Develop and demonstrate a HTML file that includes JavaScript for finding prime numbers in given range.

Software Required :V S Code .Chrome browser

Theory :

1) **for** statemen*t*

A for loop repeats until a specified condition evaluates to false. The JavaScript for loop is similar to the Java and C for loop.

When a for loop executes, the following occurs:

1. The initializing expression initialExpression, if any, is executed. This expression usually initializes one or more loop counters, but the syntax allows an expression of any degree of complexity. This expression can also declare variables.
2. The conditionExpression expression is evaluated. If the value of conditionExpression is true, the loop statements execute. If the value of condition is false, the for loop terminates. (If the condition expression is omitted entirely, the condition is assumed to be true.)
3. The statement executes. To execute multiple statements, use a block statement ({ ... }) to group those statements.
4. If present, the update expression incrementExpression is executed.
5. Control returns to Step 2.

2) ***do...while*** *statement*

- The do…while statement repeats until a specified condition evaluates to false.
- *statement* is always executed once before the condition is checked. (To execute multiple statements,

use a block statement ({ ... }) to group those statements.)

- If condition is true, the statement executes again. At the end of every execution, the condition is checked. When the condition is false, execution stops, and control passes to the statement following do...while.

1) *while* ***statement***

A while statement executes its statements as long as a specified condition evaluates to true.

If the *condition* becomes false, statement within the loop stops executing and control passes to the statement following the loop.

The condition test occurs before statement in the loop is executed. If the condition returns true, statement is executed and the *condition* is tested again. If the condition returns false, execution stops, and control is passed to the statement following while.

To execute multiple statements, use a block statement ({ ... }) to group those statements.

2) labeled *statement*

A label provides a statement with an identifier that lets you refer to it elsewhere in your program. For example, you can use a label to identify a loop, and then use the break or continue statements to indicate whether a program should interrupt the loop or continue its execution.

The value of label may be any JavaScript identifier that is not a reserved word. The statement that you identify with a label may be any statement.

3) break *statement*

Use the break statement to terminate a loop, switch, or in conjunction with a labeled statement.

- When you use break without a label, it terminates the innermost enclosing while, do-while, for, or switch immediately and transfers control to the following statement.
- When you use break with a label, it terminates the specified labeled statement.

1. The first form of the syntax terminates the innermost enclosing loop or switch.
2. The second form of the syntax terminates the specified enclosing labeled statement.

4) continue *statement*

The continue statement can be used to restart a while, do-while, for, or label statement.

- When you use continue without a label, it terminates the current iteration of the innermost enclosing while, do-while, or for statement and continues execution of the loop with the next iteration. In contrast to the break statement, continue does not terminate the execution of the loop entirely. In a while loop, it jumps back to the condition. In a for loop, it jumps to the increment-expression.

- When you use continue with a label, it applies to the looping statement identified with that label.

5) ***for...in*** *statement*

The for…in statement iterates a specified variable over all the enumerable properties of an object. For each distinct property, JavaScript executes the specified statements.

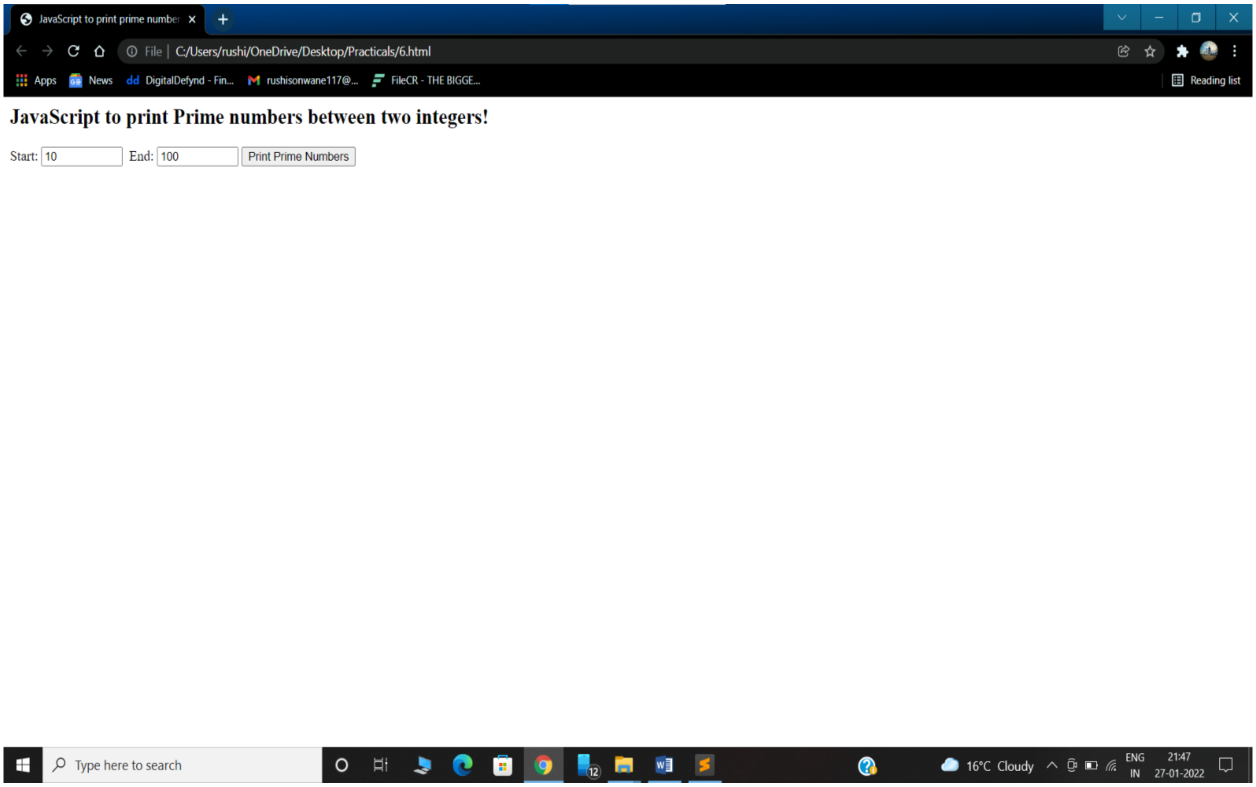
Conclusion:

1. **HTML** to define the content of web pages
2. **CSS** o specify the layout of web pages
3. **JavaScript** to program the behavior of web pages
4. JavaScript is the world's most popular programming language.
5. JavaScript is the programming language of the Web.
6. JavaScript is easy to learn.
7. JavaScript is one of the **3 languages** all web developers **must** learn:

CODE:

```
<html>
    <head>
        <title>JavaScript to print prime numbers between two
integers!</title>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-
scale=1.0">
        <script>
            function printPrime() {
                "use strict";
                 var i, j, c, start, end;
                 //get the start and end value from form
                start = parseInt(document.getElementById('start').value);
                end = parseInt(document.getElementById('end').value);
                 //clear the result div
                document.getElementById("result").innerHTML = '';
                 //loop till i equals to end
                for (i = start; i <= end; i++) {
                    c = 0;
                    for (j = 1; j <= i; j++) {
                        // % modules will give the reminder value, so if the
                             reminder is 0 then it is divisible
                        if (i % j == 0) {
                            //increment the value of c
                            c++;
                        }
                    }
                     //if the value of c is 2 then it is a prime number
                    //because a prime number should be exactly divisible by 2
times only (itself and 1)
                    if (c == 2) {
document.getElementById("result").insertAdjacentHTML('beforeend', i +
'<br>');
                    }
                }
            }
        </script>
    </head>
    <body>
        <h2>JavaScript to print Prime numbers between two integers!</h2>
        Start:  <input type="number" name="start" id="start" min="1" style="width: 100px;" value="10" />&nbsp;
        End:  <input type="number" name="end" id="end" min="1" style="width: 100px;" value="100" />&nbsp;<input type="submit" value="Print Prime Numbers" onclick="printPrime()" name="print" />
        <div id="result"></div>
    </body>
</html>
```

OUTPUT



JavaScript to print Prime numbers between two integers!

Start: 10 End: 100 Print Prime Numbers

Shri Vile Parle Kelavani Mandal's

INSTITUTE OF TECHNOLOGY

DHULE (M.S.)

DEPARMENT OF COMPUTER ENGINEERING

Subject : Web Technology Lab **Subject Code :** BTCOS407

Class: S. Y. Comp. Engg. **Expt. No. :** 07

Title: PHP program to display a digital clock which displays the current time of the server

Aim: Write a PHP program to display a digital clock which displays the current time of the server

Objective:
1.To understand the use of the PHP in web development.
2.Implementing various tasks performing in PHP
3.Run a PHP program to display a digital clock which displays the current time of the server.

Theory:

PHP is a server scripting language, and a powerful tool for making dynamic and interactive Web pages.

PHP is a widely-used, free, and efficient alternative to competitors such as Microsoft's ASP.

What is PHP?

- PHP is an acronym for "PHP: Hypertext Preprocessor"
- PHP is a widely-used, open source scripting language
- PHP scripts are executed on the server
- PHP is free to download and use

PHP is an amazing and popular language!

It is powerful enough to be at the core of the biggest blogging system on the web (WordPress)!
It is deep enough to run large social networks!
It is also easy enough to be a beginner's first server side language!

What is a PHP File?

- PHP files can contain text, HTML, CSS, JavaScript, and PHP code
- PHP code is executed on the server, and the result is returned to the browser

as plain HTML

- PHP files have extension ".php"

What Can PHP Do?

- PHP can generate dynamic page content
- PHP can create, open, read, write, delete, and close files on the server
- PHP can collect form data
- PHP can send and receive cookies
- PHP can add, delete, modify data in your database
- PHP can be used to control user-access
- PHP can encrypt data

With PHP you are not limited to output HTML. You can output images or PDF files. You can also output any text, such as XHTML and XML.

Why PHP?

- PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)
- PHP is compatible with almost all servers used today (Apache, IIS, etc.)
- PHP supports a wide range of databases
- PHP is free. Download it from the official PHP resource: www.php.net
- PHP is easy to learn and runs efficiently on the server side

"Hello World" Script in PHP

To get a feel for PHP, first start with simple PHP scripts. Since "Hello, World!" is an essential example, first we will create a friendly little "Hello, World!" script.

As mentioned earlier, PHP is embedded in HTML. That means that in amongst your normal HTML (or XHTML if you're cutting-edge) you'll have PHP statements like this –

```
<html>

  <head>
    <title>Hello World</title>
  </head>

  <body>
    <?php echo "Hello, World!";?>
  </body>

</html>
```

It will produce following result −

```
Hello, World!
```

If you examine the HTML output of the above example, you'll notice that the PHP code is not present in the file sent from the server to your Web browser. All of the PHP present in the Web page is processed and stripped from the page; the only thing returned to the client from the Web server is pure HTML output.

All PHP code must be included inside one of the three special markup tags ATE are recognised by the PHP Parser.

```
<?php PHP code goes here ?>

<?    PHP code goes here ?>

<script language = "php"> PHP code goes here </script>
```

A most common tag is the <?php...?> and we will also use the same tag in our tutorial.

From the next chapter we will start with PHP Environment Setup on your machine and then we will dig out almost all concepts related to PHP to make you comfortable with the PHP language.

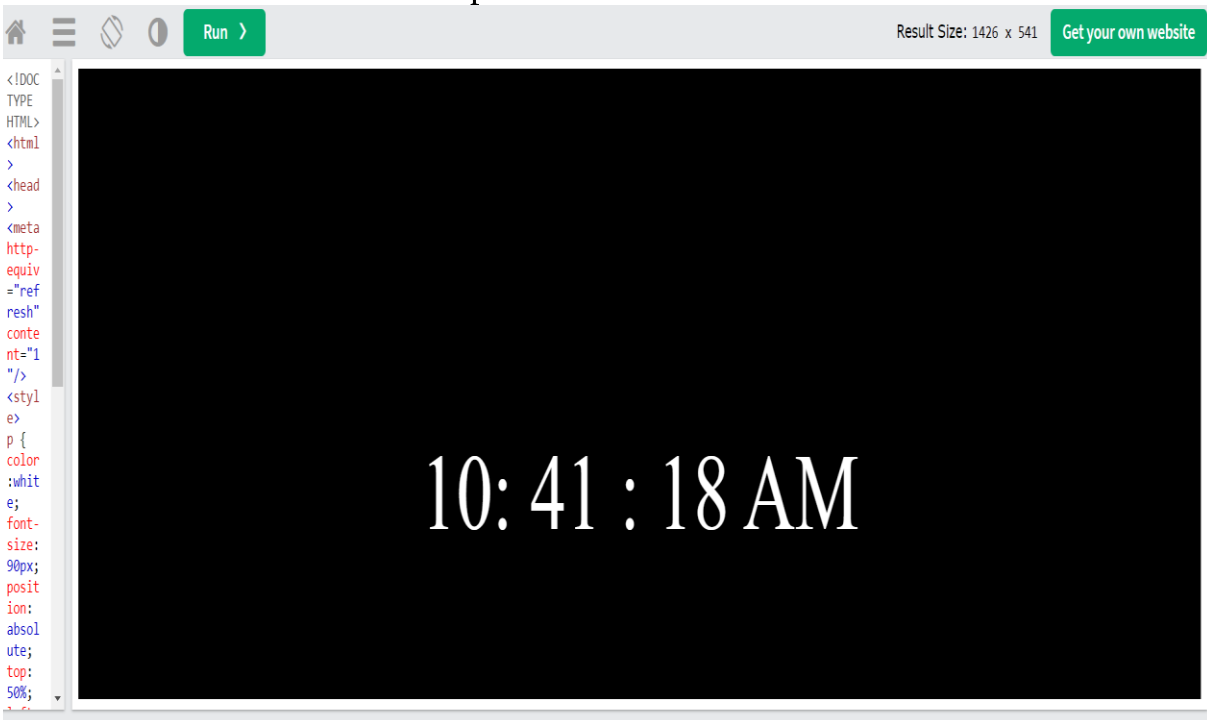
To** Display Current Running Clock in PHP**, we use a PHP script where we set the default time zone and time, a jQuery code to take that response from the PHP script then show it to the web page.

You can say it partially display running time or live clock use PHP. Here we create files, one is where we code HTML and JS then a PHP script to show time.

Code:

```
<!DOCTYPE HTML>

<html>

<head>

<meta http-equiv="refresh" content="1"/>

<style>

p {

color:white;

font-size:90px;
position: absolute;

top: 50%;

left: 50%;

transform: translate(-50%, -50%);

}

body{background-color:black;}

</style>

<p> <?php echo date(" h: i : s A");?> </p>

</head>
```

*********************Output*********************



```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="refresh" content="1"/>
<style>
p {
color:white;
font-size:90px;
position:absolute;
top:50%;
```

```
10: 41 : 18 AM
```

Shri Vile Parle Kelavani Mandal's

INSTITUTE OF TECHNOLOGY

DHULE (M.S.)

DEPARMENT OF COMPUTER ENGINEERING

Subject : Web Technology Lab **Subject Code :** BTCOS407

Class: S. Y. Comp. Engg. **Expt. No. :** 08

Title: PHP program to implement sign-In and Sign-out functionality

Aim: Write a PHP program to implement sign-In and Sign-out functionality

Objective:
1.To understand the use of the PHP in web development.
2.Implementing various tasks performing in PHP
3.Run a PHP program to implement the sign-in and sign-out functionality.

Theory:

SQL is a standard language for accessing and manipulating databases.

What is SQL?

- SQL stands for Structured Query Language
- SQL lets you access and manipulate databases
- SQL became a standard of the American National Standards Institute (ANSI) in 1986, and of the International Organization for Standardization (ISO) in 1987

What Can SQL do?

- SQL can execute queries against a database
- SQL can retrieve data from a database
- SQL can insert records in a database
- SQL can update records in a database
- SQL can delete records from a database
- SQL can create new databases
- SQL can create new tables in a database
- SQL can create stored procedures in a database
- SQL can create views in a database
- SQL can set permissions on tables, procedures, and views

SQL is a Standard - BUT....

Although SQL is an ANSI/ISO standard, there are different versions of the SQL language.However, to be compliant with the ANSI standard, they all support at least the major commands (such as SELECT, UPDATE, DELETE, INSERT, WHERE) in a similar manner.

Using SQL in Your Web Site

To build a web site that shows data from a database, you will need:

- An RDBMS database program (i.e. MS Access, SQL Server, MySQL)
- To use a server-side scripting language, like PHP or ASP
- To use SQL to get the data you want
- To use HTML / CSS to style the page

RDBMS

RDBMS stands for Relational Database Management System.

RDBMS is the basis for SQL, and for all modern database systems such as MS SQL Server, IBM DB2, Oracle, MySQL, and Microsoft Access.

The data in RDBMS is stored in database objects called tables. A table is a collection of related data entries and it consists of columns and rows.

Every table is broken up into smaller entities called fields. The fields in the Customers table consist of CustomerID, CustomerName, ContactName, Address, City, PostalCode and Country. A field is a column in a table that is designed to maintain specific information about every record in the table.

A record, also called a row, is each individual entry that exists in a table. For example, there are 91 records in the above Customers table. A record is a horizontal entity in a table.

A column is a vertical entity in a table that contains all information associated with a specific field in a table.

Some of The Most Important SQL Commands

- SELECT - extracts data from a database
- UPDATE - updates data in a database
- DELETE - deletes data from a database

- INSERT INTO - inserts new data into a database
- CREATE DATABASE - creates a new database
- ALTER DATABASE - modifies a database
- CREATE TABLE - creates a new table
- ALTER TABLE - modifies a table
- DROP TABLE - deletes a table
- CREATE INDEX - creates an index (search key)
- DROP INDEX - deletes an index

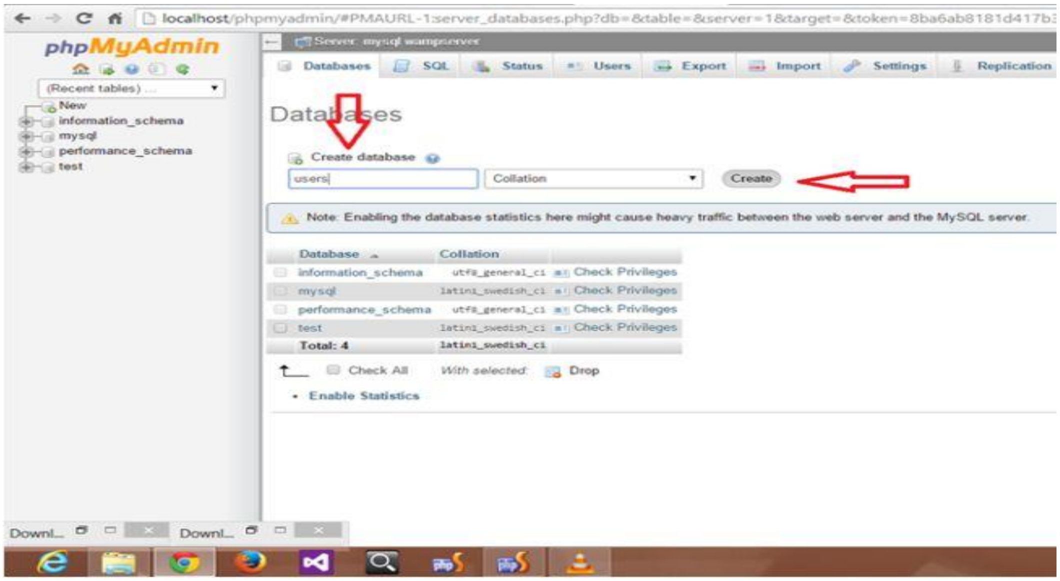
The following are the points we will be learning in this practicel,

1. Creating a Database in PHPMyAdmin.
2. Create a connection with the MySQL database.
3. Insert, delete, and view data from MySQL database.
4. Some Bootstrap components.
5. Sessions in PHP.

Files and IDE Registration

1. Login
2. Logout
3. Welcome
4. Admin_login
5. View_users
6. Db_conection
7. Delete.php

Create Database

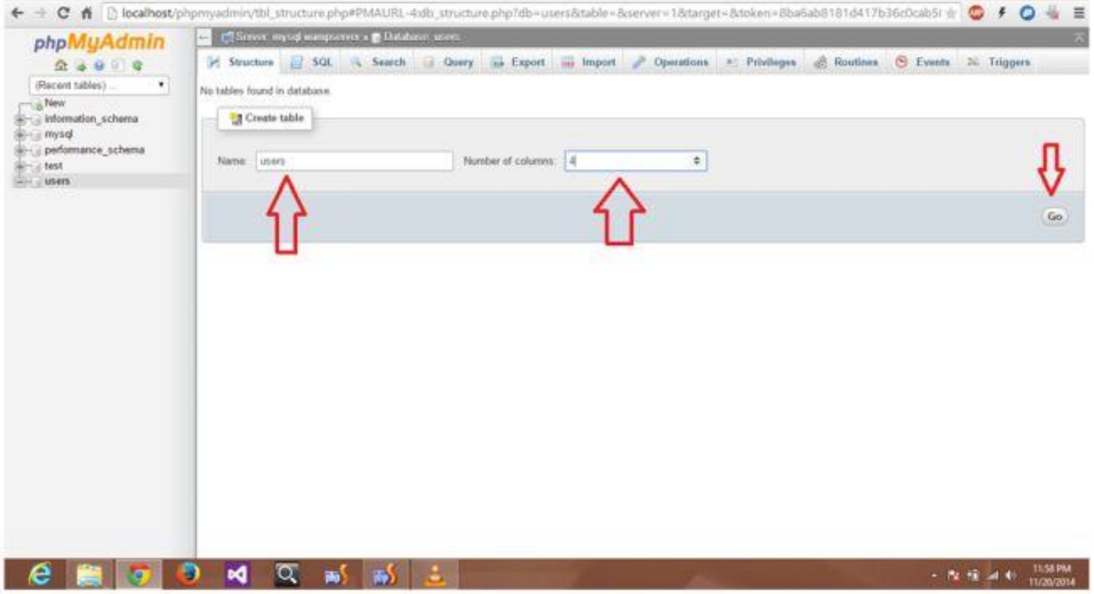


Create database: users

Note: Enabling the database statistics here might cause heavy traffic between the web server and the MySQL server.

Database	Collation	
information_schema	utf8_general_ci	Check Privileges
mysql	latin1_swedish_ci	Check Privileges
performance_schema	utf8_general_ci	Check Privileges
test	latin1_swedish_ci	Check Privileges
Total: 4	latin1_swedish_ci	

Create Tables

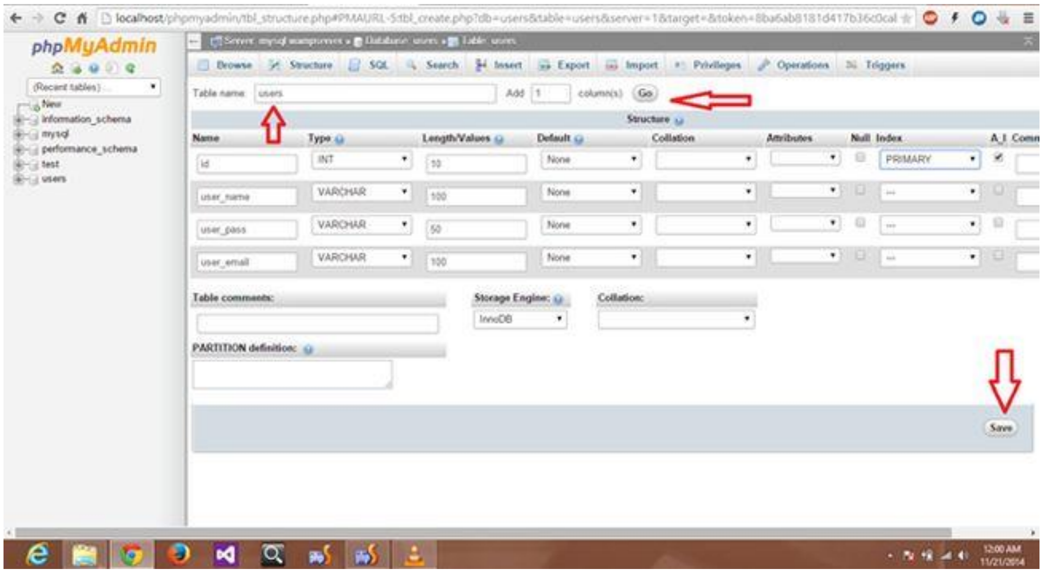


No tables found in database.

Name: users

Number of columns: 4

Create columns in the Users table



localhost/phpmyadmin/tbl_structure.php#PMAURL-5:tbl_create.php?db=users&table=users&server=1&target=&token=8ba6ab8181d417b36c0cal

Server: mysql wampserver » Database: users » Table: users

Table name: users   Add 1 column(s) Go

Structure

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	A_I
id	INT	10	None			□	PRIMARY	☑
user_name	VARCHAR	100	None			□	---	□
user_pass	VARCHAR	50	None			□	---	□
user_email	VARCHAR	100	None			□	---	□

Table comments:

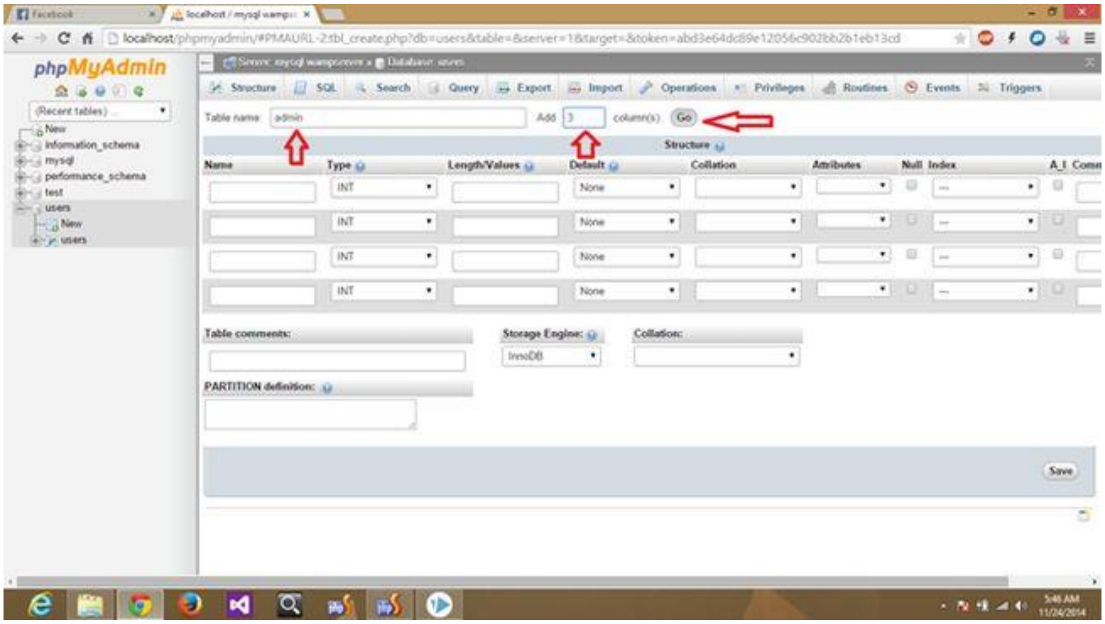
Storage Engine: InnoDB

Collation:

PARTITION definition:

Save

Create Admin Table



localhost/phpmyadmin/#PMAURL-2:tbl_create.php?db=users&table=&server=1&target=&token=abd3e64dc89e12056c902bb2b1eb13cd

Server: mysql wampserver » Database: users

Table name: admin   Add 3 column(s) Go

Structure

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	A_I
	INT		None			□	---	□
	INT		None			□	---	□
	INT		None			□	---	□
	INT		None			□	---	□

Table comments:

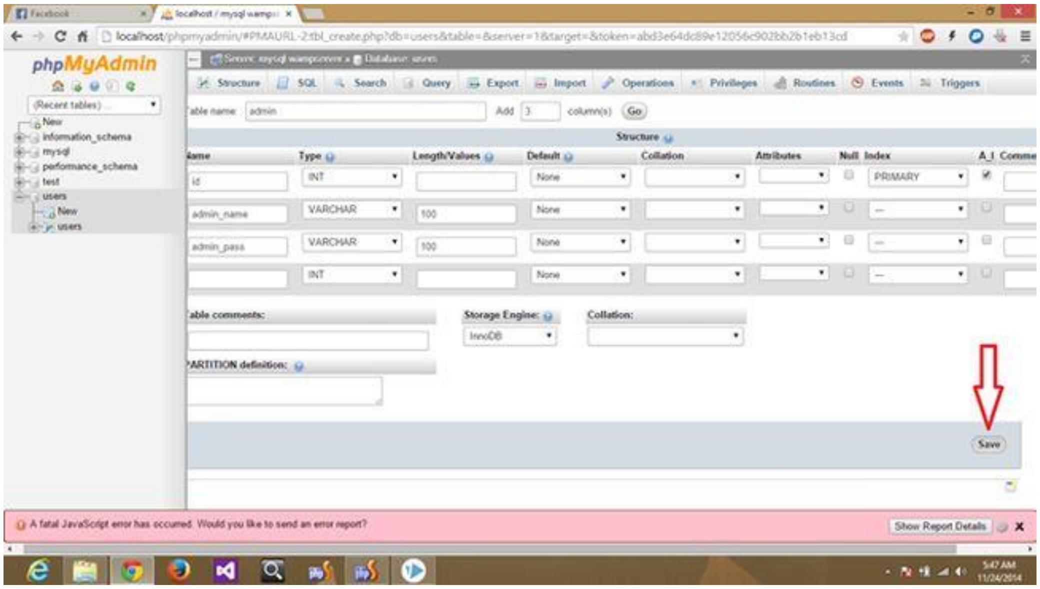
Storage Engine: InnoDB

Collation:

PARTITION definition:

Save

Create Admin Columns



localhost/phpmyadmin/#PMAURL-2:tbl_create.php?db=users&table=&server=1&target=&token=abd3e64dc89e12056c902bb2b1eb13cd

able name: admin   Add 3 column(s) Go

ame	Type	Length/Values	Default	Collation	Attributes	Null	Index	A_I
id	INT		None			□	PRIMARY	☑
admin_name	VARCHAR	100	None			□	---	□
admin_pass	VARCHAR	100	None			□	---	□
	INT		None			□	---	□

Storage Engine: InnoDB

A fatal JavaScript error has occurred. Would you like to send an error report?

Code:

PHPL Programs

1. **Registration.php**

```
        <html>
        <head lang="en">
            <meta charset="UTF-8">
    <link type="text/css" rel="stylesheet" href="bootstrap-3.2.0-dist\css\bootstrap.css">
            <title>Registration</title>
        </head>
        <style>
            .login-panel {
                margin-top: 150px;

        </style>
        <body>
        <div class="container"><!-- container class is used to centered  the body of the browser with some decent width-->
            <div class="row"><!-- row class is used for grid system in Bootstrap-->
                <div class="col-md-4 col-md-offset-4"><!--col-md-4 is used to create the no of colums in the grid also use for medimum and large devices-->
                    <div class="login-panel panel panel-success">
                        <div class="panel-heading">
                            <h3 class="panel-title">Registration</h3>
                        </div>
                        <div class="panel-body">
                            <form role="form" method="post"
action="registration.php">
                                <fieldset>
                                    <div class="form-group">
                                        <input class="form-control" placeholder="Username"
name="name" type="text" autofocus>
                                    </div>

                                    <div class="form-group">
                                        <input class="form-control" placeholder="E-mail"
name="email" type="email" autofocus>
                                    </div>
                                    <div class="form-group">
                                        <input class="form-control" placeholder="Password"
name="pass" type="password" value="">
                                    </div>
                                    <input class="btn btn-lg btn-success btn-block"
type="submit" value="register" name="register" >
                                </fieldset>
                            </form>
                            <center><b>Already registered ?</b> <br></b><a
href="login.php">Login here</a></center><!--for centered text-->
                        </div>
                    </div>
                </div>
            </div>
        </div>
        </body>
        </html>
        <?php
        include("database/db_conection.php");//make connection here
        if(isset($_POST['register']))
        {
            $user_name=$_POST['name'];//here getting result from the post array
after submitting the form.
            $user_pass=$_POST['pass'];//same
            $user_email=$_POST['email'];//same
            if($user_name==")
            {
                //javascript use for input checking
                echo"<script>alert('Please enter the name')</script>";
        exit();//this use if first is not work then other will not show
```

```
            }
            if($user_pass=='')
            {
               echo"<script>alert('Please enter the password')</script>";
        exit();
            }
            if($user_email=='')
            {
               echo"<script>alert('Please enter the email')</script>";
            exit();
            }
        //here query check weather if user already registered so can't register
again.
            $check_email_query="select * from users WHERE
user_email='$user_email'";
            $run_query=mysqli_query($dbcon,$check_email_query);
            if(mysqli_num_rows($run_query)>0)
            {
        echo "<script>alert('Email $user_email is already exist in our database,
Please try another one!')</script>";
        exit();
            }
        //insert the user into the database.
            $insert_user="insert into users (user_name,user_pass,user_email)
VALUE ('$user_name','$user_pass','$user_email')";
            if(mysqli_query($dbcon,$insert_user))
            {
               echo"<script>window.open('welcome.php','_self')</script>";
            }
        }
        ?>
```

2. **Login.php**

```
        <?php
        session_start();//session starts here

        ?>

        <html>
        <head lang="en">
            <meta charset="UTF-8">
            <link type="text/css" rel="stylesheet" href="bootstrap-3.2.0-
dist\css\bootstrap.css">
```

```
    <title>Login</title>
</head>
<style>
    .login-panel {
        margin-top: 150px;
</style>
<body>
<div class="container">
   <div class="row">
        <div class="col-md-4 col-md-offset-4">
            <div class="login-panel panel panel-success">
                <div class="panel-heading">
                    <h3 class="panel-title">Sign In</h3>
                </div>
                <div class="panel-body">
                    <form role="form" method="post" action="login.php">
                        <fieldset>
                            <div class="form-group"  >
                                <input class="form-control" placeholder="E-mail" name="email" type="email" autofocus>
                            </div>
                            <div class="form-group">
                                <input class="form-control" placeholder="Password" name="pass" type="password" value="">
                            </div>
                                <input class="btn btn-lg btn-success btn-block" type="submit" value="login" name="login" >
                            <!-- Change this to a button or input when using this as a form -->
                          <!--  <a href="index.html" class="btn btn-lg btn-success btn-block">Login</a> -->
                        </fieldset>
                    </form>
                </div>
            </div>
        </div>
    </div>
</div>
</body>
</html>
<?php
include("database/db_conection.php");
```

```
        if(isset($_POST['login']))
        {
            $user_email=$_POST['email'];
            $user_pass=$_POST['pass'];
            $check_user="select * from users WHERE user_email='$user_email'AND user_pass='$user_pass'";
            $run=mysqli_query($dbcon,$check_user);
            if(mysqli_num_rows($run))
            {
                echo "<script>window.open('welcome.php','_self')</script>";
                $_SESSION['email']=$user_email;//here session is used and value of $user_email store in $_SESSION.
            }
            else
            {
              echo "<script>alert('Email or password is incorrect!')</script>";
            }
        }
        ?>
```

3. Logout.php

```
        <?php
        /**
         * Created by PhpStorm.
         * User: Ehtesham Mehmood
         * Date: 11/21/2014
         * Time: 2:46 AM
         */
        session_start();//session is a way to store information (in variables) to be used across multiple pages.
        session_destroy();
        header("Location: login.php");//use for the redirection to some page
        ?>
```

4. Welcome.php

```
        <?php
        session_start();

        if(!$_SESSION['email'])
        {
            header("Location: login.php");//redirect to the login page to secure the welcome page without login access.
        }
        ?>
        <html>
        <head>
            <title>
                Registration
            </title>
        </head>
        <body>
        <h1>Welcome</h1><br>
        <?php
        echo $_SESSION['email'];
        ?>
        <h1><a href="logout.php">Logout here</a> </h1>
        </body>
        </html>
```

5. Admin_login.php

```
        <html>
        <head lang="en">
           <meta charset="UTF-8">
           <link type="text/css" rel="stylesheet" href="bootstrap-3.2.0-
```

```
dist\css\bootstrap.css">
            <title>Admin Login</title>
        </head>
        <style>
            .login-panel {
                margin-top: 150px;
        </style>
        <body>
        <div class="container">
            <div class="row">
                <div class="col-md-4 col-md-offset-4">
                    <div class="login-panel panel panel-success">
                        <div class="panel-heading">
                            <h3 class="panel-title">Sign In</h3>
                        </div>
                        <div class="panel-body">
                            <form role="form" method="post" action="admin_login.php">
                                <fieldset>
                                    <div class="form-group"  >
                                        <input class="form-control" placeholder="Name" name="admin_name" type="text" autofocus>
                                    </div>
                                    <div class="form-group">
                                        <input class="form-control" placeholder="Password" name="admin_pass" type="password" value="">
                                    </div>
                                    <input class="btn btn-lg btn-success btn-block" type="submit" value="login" name="admin_login" >
                                </fieldset>
                            </form>
                        </div>
                    </div>
                </div>
```

```
    </div>
</div>
</body>
</html>
<?php
/**
 * Created by PhpStorm.
 * User: Ehtesham Mehmood
 * Date: 11/24/2014
 * Time: 3:26 AM
 */
include("database/db_conection.php");

if(isset($_POST['admin_login']))//this will tell us what to do if some data has been post through form with button.
{
    $admin_name=$_POST['admin_name'];
    $admin_pass=$_POST['admin_pass'];
    $admin_query="select * from admin where admin_name='$admin_name' AND admin_pass='$admin_pass'";
    $run_query=mysqli_query($dbcon,$admin_query);
    if(mysqli_num_rows($run_query)>0)
    {
        echo "<script>window.open('view_users.php','_self')</script>";
    }
    else {echo"<script>alert('Admin Details are incorrect..!')</script>";}
}
?>
```

6. View_users.php

```
<html>
<head lang="en">
    <meta charset="UTF-8">
    <link type="text/css" rel="stylesheet" href="bootstrap-3.2.0-
```

```
dist\css\bootstrap.css"> <!--css file link in bootstrap folder-->
        <title>View Users</title>
      </head>
      <style>
        .login-panel {
          margin-top: 150px;
        }
        .table {
          margin-top: 50px;
         }
      </style>
      <body>
      <div class="table-scrol">
        <h1 align="center">All the Users</h1>
      <div class="table-responsive"><!--this is used for responsive display in mobile and other devices-->
        <table class="table table-bordered table-hover table-striped" style="table-layout: fixed">
          <thead>
          <tr>
            <th>User Id</th>
            <th>User Name</th>
            <th>User E-mail</th>
            <th>User Pass</th>
            <th>Delete User</th>
          </tr>
          </thead>
          <?php
          include("database/db_conection.php");
          $view_users_query="select * from users";//select query for viewing users.
          $run=mysqli_query($dbcon,$view_users_query);//here run the sql query.
          while($row=mysqli_fetch_array($run))//while look to fetch the
```

```
result and store in a array $row.
                {
                   $user_id=$row[0];
                   $user_name=$row[1];
                   $user_email=$row[2];
                   $user_pass=$row[3];
                ?>
                <tr>
          <!--here showing results in the table -->
                   <td><?php echo $user_id;  ?></td>
                   <td><?php echo $user_name;  ?></td>
                   <td><?php echo $user_email;  ?></td>
                   <td><?php echo $user_pass;  ?></td>
                   <td><a href="delete.php?del=<?php echo $user_id ?>"><button
class="btn btn-danger">Delete</button></a></td> <!--btn btn-danger is a
bootstrap button to show danger-->
                </tr>
                <?php } ?>
             </table>
                </div>
          </div>
          </body>
          </html>
```

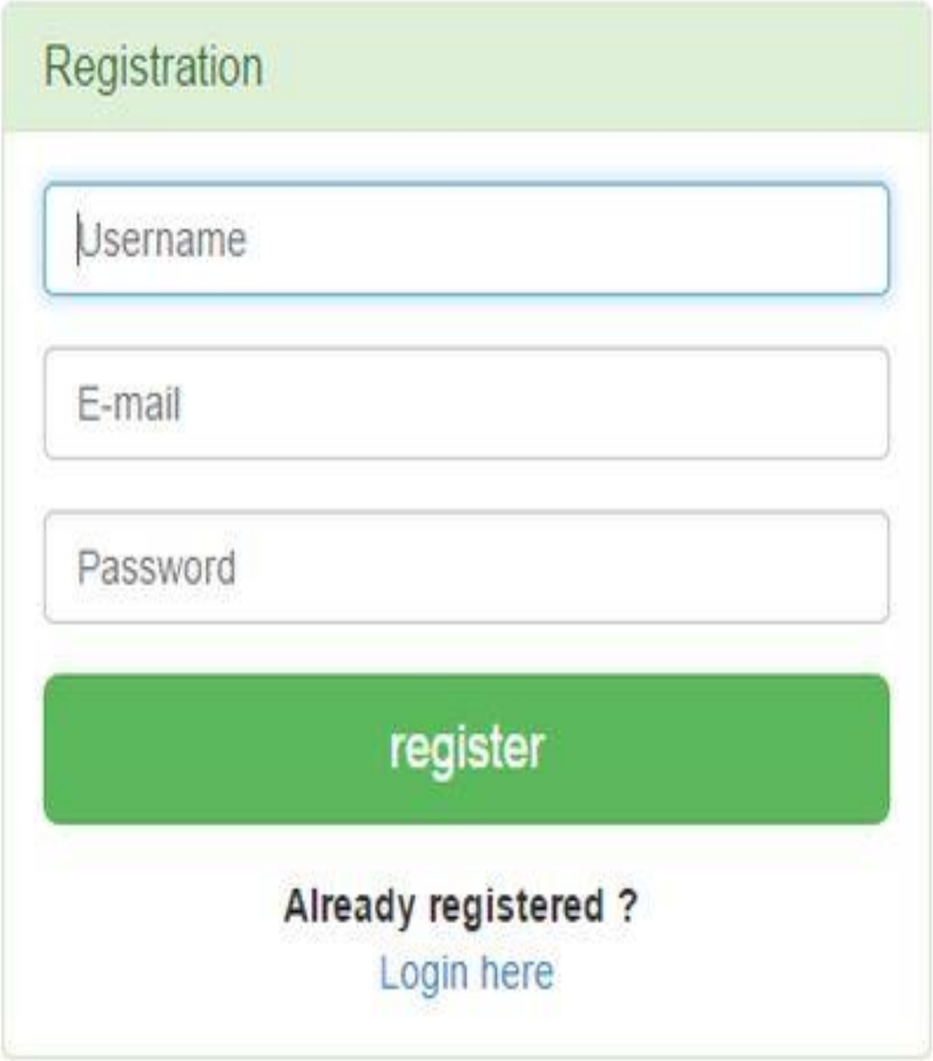
7. Db_conection.php

```
          <?php
          /**
           * Created by PhpStorm.
           * User: Ehtesham Mehmood
           * Date: 11/21/2014
           * Time: 1:13 AM
           */
          $dbcon=mysqli_connect("localhost","root","");
          mysqli_select_db($dbcon,"users");
```

```
        ?>
```

8. Delete.php

```
        <?php
        /**
         * Created by PhpStorm.
         * User: Ehtesham Mehmood
         * Date: 11/24/2014
         * Time: 11:55 PM
         */
        include("database/db_conection.php");
        $delete_id=$_GET['del'];
        $delete_query="delete  from users WHERE id='$delete_id'";//delete query
        $run=mysqli_query($dbcon,$delete_query);
        if($run)
        {
        //javascript function to open in the same window
 echo "<script>window.open('view_users.php?deleted=user has been deleted','_self')</script>";
        }
        ?>
```

	************OUTPUT**************  Registration Username E-mail Password register Already registered ? Login here

Shri Vile Parle Kelavani Mandal's

INSTITUTE OF TECHNOLOGY

DHULE (M.S.)

DEPARMENT OF COMPUTER ENGINEERING

Subject : Web Technology Lab **Subject Code :** BTCOS407

Class: S. Y. Comp. Engg. **Expt. No. :** 09

Title: PHP program to keep track of the number of visitors visiting the Web page and to display this count of visitors, with proper headings.

Aim: Write a PHP program to keep track of the number of visitors visiting the Web page and to display this count of visitors, with proper headings.

Objective: 1.To understand the use of the PHP in web development.
2.Implementing various tasks performing in PHP
3.Run a PHP program to keep track of the number of visitors visiting the Web page and to display this count of visitors, with proper headings.

Theory:

What is a session?

A PHP session is used to store data on a server rather than the computer of the user.Session identifiers or SID is unique numbers which are used to identify every user in a session based environment.The SID is used to link the user with his information on the server like posts, emails etc. You can learn about sessions in details in the article PHP | Sessions

How to use sessions for Storing Page Counts

A session mechanism can be used to store page views which increment on each refresh and show the count on a webpage. A session is user specific and for every user, a separate session is created along with a separate session variable which is associated with that session.
Using this mechanism, for every user the session variable is set to 1 initially for the first visit.On consecutive visits, the value of this session variable is incremented and displayed on the output webpage.

Below is the explanation of above code:

1. **session_start() :**It is a first step which is used to start the session. It is a standard call. The session_start() should be used whenever the session variable is used.
2. **$_SESSION['views'] :**This is the session variable which is used to store views count for a user's session. 'views' is the session name. The session name should be always be enclosed within the single quote.

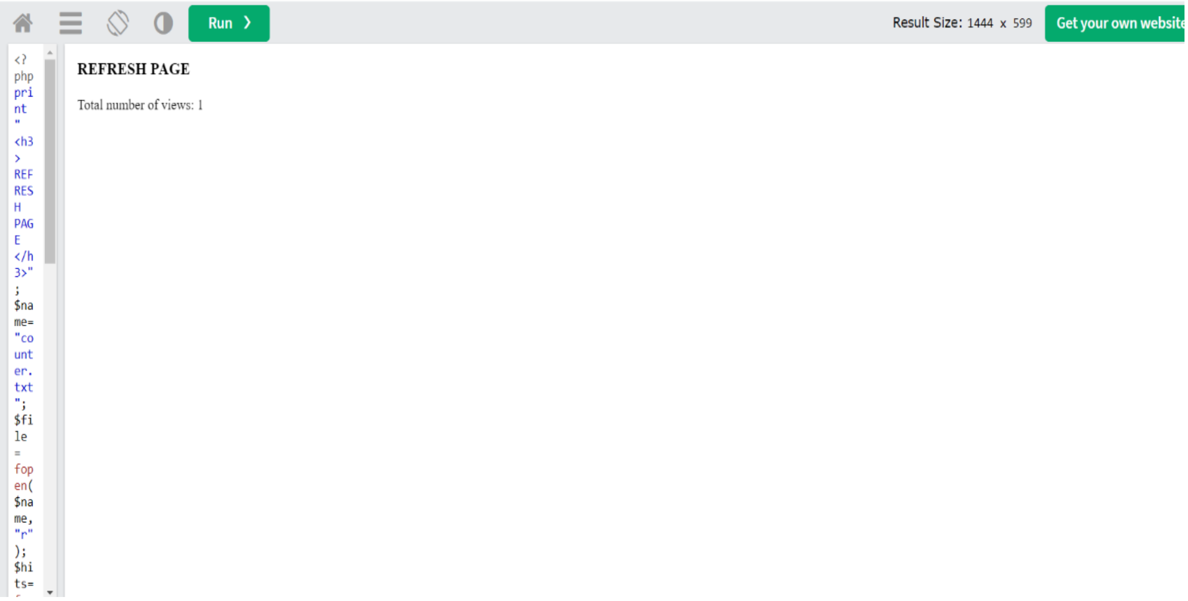
3. **isset() :** It is a standard php function which returns true or false depending upon whether the passed parameter is set or not.

Code:

Code:

```
<?php
print "<h3> REFRESH PAGE </h3>";
$name="counter.txt";
$file = fopen($name,"r");
$hits= fscanf($file,"%d");
fclose($file);
$hits[0]++;
$file = fopen($name,"w");
fprintf($file,"%d",$hits[0]);
fclose($file);
print "Total number of views: ".$hits[0];
?>
```

Output:



```
<?php
print "<h3> REFRESH PAGE </h3>";
$name="counter.txt";
$file = fopen($name,"r");
$hits=
```

REFRESH PAGE

Total number of views: 1

Shri Vile Parle Kelavani Mandal's

INSTITUTE OF TECHNOLOGY

DHULE (M.S.)

DEPARMENT OF COMPUTER ENGINEERING

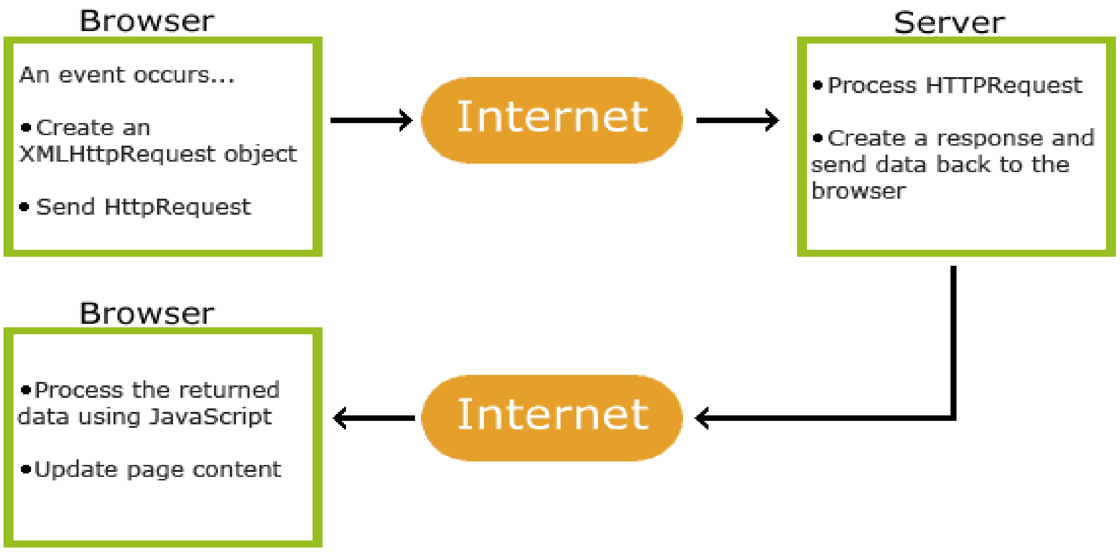
Subject : Web Technology Lab **Subject Code :** BTCOS407

Class: S. Y. Comp. Engg. **Expt. No. :** 10

Title: Write a PHP code to implement AJAX functionality.

Aim:	Write a PHP code to implement AJAX functionality.
Objective:	1.To understand the use of the PHP in web development. 2.Implementing various tasks performing in PHP 3.Run a PHP code to implement AJAX functionality.
Theory:	**AJAX Introduction:** AJAX is about updating parts of a web page, without reloading the whole page. AJAX = Asynchronous JavaScript and XML. AJAX is a technique for creating fast and dynamic web pages. AJAX allows web pages to be updated asynchronously by exchanging small amounts of data with the server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page. Classic web pages, (which do not use AJAX) must reload the entire page if the content should change. Examples of applications using AJAX: Google Maps, Gmail, Youtube, and Facebook tabs.

How AJAX Works


Browser
An event occurs...
• Create an XMLHttpRequest object
• Send HttpRequest
Internet
Server
• Process HTTPRequest
• Create a response and send data back to the browser
Browser
• Process the returned data using JavaScript
• Update page content
Internet

AJAX is Based on Internet Standards

AJAX is based on internet standards, and uses a combination of:

- XMLHttpRequest object (to exchange data asynchronously with a server)
- JavaScript/DOM (to display/interact with the information)
- CSS (to style the data)
- XML (often used as the format for transferring data)

AJAX applications are browser- and platform-independent!

PHP - AJAX and PHP

AJAX is used to create more interactive applications.

Code:

Code

First, check if the input field is empty (str.length == 0). If it is, clear the content of the txtHint placeholder and exit the function.

However, if the input field is not empty, do the following:

- Create an XMLHttpRequest object
- Create the function to be executed when the server response is ready

- Send the request off to a PHP file (gethint.php) on the server
- Notice that q parameter is added to the url (gethint.php?q="+str)
- And the str variable holds the content of the input field

CODE:

```
<html>
<head>
<script>
function showHint(str) {
  if (str.length == 0) {
    document.getElementById("txtHint").innerHTML = "";
    return;
  } else {
    var xmlhttp = new XMLHttpRequest();
    xmlhttp.onreadystatechange = function() {
      if (this.readyState == 4 && this.status == 200) {
        document.getElementById("txtHint").innerHTML = this.responseText;
      }
    };
    xmlhttp.open("GET", "gethint.php?q=" + str, true);
    xmlhttp.send();
  }
}
</script>
</head>
<body>

<p><b>Start typing a name in the input field below:</b></p>
<form action="">
  <label for="fname">First name:</label>
  <input type="text" id="fname" name="fname" onkeyup="showHint(this.value)">
</form>
<p>Suggestions: <span id="txtHint"></span></p>
</body>
</html>
```

The PHP File - "gethint.php"

The PHP file checks an array of names, and returns the corresponding name(s) to the browser:

```php
<?php
// Array with names
$a[] = "Anna";
$a[] = "Brittany";
$a[] = "Cinderella";
$a[] = "Diana";
$a[] = "Eva";
$a[] = "Fiona";
$a[] = "Gunda";
$a[] = "Hege";
$a[] = "Inga";
$a[] = "Johanna";
$a[] = "Kitty";
$a[] = "Linda";
$a[] = "Nina";
$a[] = "Ophelia";
$a[] = "Petunia";
$a[] = "Amanda";
$a[] = "Raquel";
$a[] = "Cindy";
$a[] = "Doris";
$a[] = "Eve";
$a[] = "Evita";
$a[] = "Sunniva";
$a[] = "Tove";
$a[] = "Unni";
$a[] = "Violet";
$a[] = "Liza";
$a[] = "Elizabeth";
$a[] = "Ellen";
$a[] = "Wenche";
$a[] = "Vicky";

// get the q parameter from URL
$q = $_REQUEST["q"];

$hint = "";

// lookup all hints from array if $q is different from ""
if ($q !== "") {
  $q = strtolower($q);
  $len=strlen($q);
  foreach($a as $name) {
    if (stristr($q, substr($name, 0, $len))) {
      if ($hint === "") {
        $hint = $name;
      } else {
        $hint .= ", $name";
      }
    }
  }
}

// Output "no suggestion" if no hint was found or output correct values
echo $hint === "" ? "no suggestion" : $hint;
?>
```

Shri Vile Parle Kelavani Mandal's

INSTITUTE OF TECHNOLOGY

DHULE (M.S.)

DEPARMENT OF COMPUTER ENGINEERING

Subject : Web Technology Lab **Subject Code :** BTCOS407

Class: S. Y. Comp. Engg. **Expt. No. :** 11

Title: Write a PHP program to perform search operation on the student records using AJAX.

Aim: Write a PHP program to perform search operation on the student records using AJAX

Objective:
1.To understand the use of the PHP in web development.
2.Implementing various tasks performing in PHP
3.Run a PHP code to perform search operation on the student records using AJAX

Theory:

PHP - AJAX and MySQL

AJAX Database Example

The following example will demonstrate how a web page can fetch information from a database with AJAX:

ExampleGet your own PHP Server

Example Explained - The MySQL Database

The database table we use in the example above looks like this:

id	FirstName	LastName	Age	Hometown	PRN_no.
1	Peter	Griffin	41	Quahog	111112222

2	Lois	Griffin	40	Newport	1111133333
3	Joseph	Swanson	39	Quahog	11114444
4	Glenn	Quagmire	41	Quahog	1111115555

Example Explained

In the example above, when a user selects a person in the dropdown list above, a function called "showUser()" is executed.

The function is triggered by the onchange event.

Here is the HTML code:

Example

```
<html>
<head>
<script>
function showUser(str) {
  if (str == "") {
    document.getElementById("txtHint").innerHTML = "";
    return;
  } else {
    var xmlhttp = new XMLHttpRequest();
    xmlhttp.onreadystatechange = function() {
      if (this.readyState == 4 && this.status == 200) {
        document.getElementById("txtHint").innerHTML = this.responseText;
      }
    };
    xmlhttp.open("GET","getuser.php?q="+str,true);
    xmlhttp.send();
  }
}
</script>
```

```
</head>
<body>

<form>
<select name="users" onchange="showUser(this.value)">
  <option value="">Select a person:</option>
  <option value="1">Peter Griffin</option>
  <option value="2">Lois Griffin</option>
  <option value="3">Joseph Swanson</option>
  <option value="4">Glenn Quagmire</option>
  </select>
</form>
<br>
<div id="txtHint"><b>Person info will be listed here...</b></div>

</body>
</html>
```

Run example »

Code explanation:

First, check if person is selected. If no person is selected (str == ""), clear the content of txtHint and exit the function. If a person is selected, do the following:

- Create an XMLHttpRequest object
- Create the function to be executed when the server response is ready
- Send the request off to a file on the server
- Notice that a parameter (q) is added to the URL (with the content of the dropdown list)

The PHP File

The page on the server called by the JavaScript above is a PHP file called "getuser.php".

The source code in "getuser.php" runs a query against a MySQL database, and returns the result in an HTML table:

```
<!DOCTYPE html>
<html>
```

```
<head>
<style>
table {
  width: 100%;
  border-collapse: collapse;
}

table, td, th {
  border: 1px solid black;
  padding: 5px;
}

th {text-align: left;}
</style>
</head>
<body>

<?php
$q = intval($_GET['q']);

$con = mysqli_connect('localhost','peter','abc123');
if (!$con) {
  die('Could not connect: ' . mysqli_error($con));
}

mysqli_select_db($con,"ajax_demo");
$sql="SELECT * FROM user WHERE id = '".$q."'";
$result = mysqli_query($con,$sql);

echo "<table>
<tr>
<th>Firstname</th>
<th>Lastname</th>
<th>Age</th>
<th>Hometown</th>
<th>Job</th>
</tr>";
while($row = mysqli_fetch_array($result)) {
  echo "<tr>";
  echo "<td>" . $row['FirstName'] . "</td>";
  echo "<td>" . $row['LastName'] . "</td>";
  echo "<td>" . $row['Age'] . "</td>";
  echo "<td>" . $row['Hometown'] . "</td>";
  echo "<td>" . $row['Job'] . "</td>";
  echo "</tr>";
}
echo "</table>";
mysqli_close($con);
?>
</body>
</html>
```

Explanation: When the query is sent from the JavaScript to the PHP file, the following happens:

1. PHP opens a connection to a MySQL server
2. The correct person is found

1. An HTML table is created, filled with data, and sent back to the "txtHint" placeholder

Firstname	**Lastname**	**Age**	**Hometown**	**PRN_no.**
Joseph	Swanson	39	Quahog	1111444

Shri Vile Parle Kelavani Mandal's

INSTITUTE OF TECHNOLOGY

DHULE (M.S.)

DEPARMENT OF COMPUTER ENGINEERING

Subject : Web Technology Lab **Subject Code :** BTCOS407

Class: S. Y. Comp. Engg. **Expt. No. :** 12

Title: PHP program to sort the student records which are stored in the database using ascending/descending order.

Aim: Write a PHP program to sort the student records which are stored in the database using ascending/descending order.

Objective:
1.To understand the use of the PHP in web development.
2.Implementing various tasks performing in PHP
3.Run a PHP code to sort the student records which are stored in the database using ascending/descending order.

Theory:

PHP MySQL Use The ORDER BY Clause

Select and Order Data From a MySQL Database

The ORDER BY clause is used to sort the result-set in ascending or descending order.

The ORDER BY clause sorts the records in ascending order by default. To sort the records in descending order, use the DESC keyword.

SELECT column_name(s) FROM table_name ORDER BY column_name(s) ASC|DESC

Select and Order Data With MySQLi

The following example selects the id, firstname and lastname columns from the MyGuests table. The records will be ordered by the lastname column:

Example (MySQLi Object-oriented)

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";
```

```
// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
  die("Connection failed: " . $conn->connect_error);
}

$sql = "SELECT id, firstname, lastname FROM MyGuests ORDER BY
lastname";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
  // output data of each row
  while($row = $result->fetch_assoc()) {
    echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " .
$row["lastname"]. "<br>";
  }
} else {
  echo "0 results";
}
$conn->close();
?>
```

Code lines to explain from the example above:

First, we set up the SQL query that selects the id, firstname and lastname columns from the MyGuests table. The records will be ordered by the lastname column. The next line of code runs the query and puts the resulting data into a variable called $result.

Then, the function num_rows() checks if there are more than zero rows returned.

If there are more than zero rows returned, the function fetch_assoc() puts all the results into an associative array that we can loop through. The while() loop loops through the result set and outputs the data from the id, firstname and lastname columns.

The following example shows the same as the example above, in the MySQLi procedural way:

Example (MySQLi Procedural)

```
<?php
$servername = "localhost";
```

```
$username = "username";
$password = "password";
$dbname = "myDB";

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
  die("Connection failed: " . mysqli_connect_error());
}

$sql = "SELECT id, firstname, lastname FROM MyGuests ORDER BY
lastname";
$result = mysqli_query($conn, $sql);

if (mysqli_num_rows($result) > 0) {
  // output data of each row
  while($row = mysqli_fetch_assoc($result)) {
    echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " .
$row["lastname"]. "<br>";
  }
} else {
  echo "0 results";
}

mysqli_close($conn);
?>
```