Introduction

This lab is intended to give the students a sound knowledge in the Web side programming. Before going in to the details of the lab, the pre-requisites are the basic knowledge in HTML, XHTML, CSS, XML, JavaScript, Perl, PHP and MySql. Let’s look at some of these topics in brief now.

Basics of HTML: -

Hyper Text Markup Language (HTML) is a mark up language developed by the W3C people. This can be used as an interface for working our programs. We submit all our requests in the HTML form. It is basically a markup language which describes how the documents are to be formatted.

HTML has two basic entities, the “Tags” (Formatting commands) and the strings within the tags called as the “Directives”. Most of the tags have the following syntax: - <something> that indicates the beginning of the tag and a </something> that indicates the end of the tag.

*NOTE:*

* Tags can either be in lower case or upper case, i.e. there is no difference between <html> and <HTML>
* The order in which parameters of the tag are given is not significant since each of these parameters is named.

One major *frustration* that comes with using HTML is that it does *not provide any structure to Web Pages. It also mixes the content with the formatting.*

HTML Essentials

An HTML file should be written in the following format and should be saved with .html or .htm file extension

<html>

<body>

<head>

<title> New Page </title>

</head>

TYPE YOUR TEXT HERE

</body>

</html>

The “*New Page*” title comes on the top of the Browser Window.

Basic HTML Tags: -

To create a text box

<input type=text name=T1 size=20>

To create a Normal Button

<input type=button name=B4 value=GO>

To create a Submit Button

<input type=submit name=B1 size=20>

To create a Reset Button

<input type=reset name=T1 size=20>

To create a Radio Button

<input type=radio value=V1 checked name=R1>

To create a Check box

<input type=checkbox name=C1 value=ON >

To create a Form

<form method=[GET/POST] action=[url]>

<input type=submit value=Submit name=B1>

<input type=reset value=Reset name="B2">

</form>

To create a Text Area

<textarea rows=2 name=S1 cols=20></textarea>

To create a Drop down Menu

<select size=1 name=D1></select>

To create a Hyper Link

<a href=http://localhost: 8080/a.htm>BACK </a>

To create a Marquee *(The Marquee tag ensures that the text scrolls horizontally across the screen. It is usually used by Advertisement sites to catch the user’s attention. Although they sound and look cool, it is preferred to avoid using too much of theses since they can be tiring and confusing to the users eye especially if the scroll speed is set too high)*

<marquee align=middle>Type your text here</marquee>

To give Background color

<body bgcolor=green>…</body>

*(The basics colours can be given literally here. For a more elaborate set if colours, Hex code of the colours can be given. Refer to the possible ranges of the Hex codes in a HTML Book)*

More Miscellaneous Tags: -

1. <h#>……..</h#> - where ‘#’ is a number ranging from 1-6. This is used to set the text size.
2. <pre> - Preformatted text, ensures that the text appears exactly the way it appears in the HTML code thereby preserving the white spaces as well.
3. <br> - Inserts a “New line” character (similar to ‘\n’).
4. To Draw a Horizontal Line (Horizontal Ruler):

<hr size=4 width=”50 %”>

1. < b > - Bold, < I > - Italics
2. Tables:

<table>

<caption> *Your Caption here* </caption> [Optional Tag]

<tr>

<th> *Row 1, Col 1* </th> [th implies Table Header]

<th> *Row 2, Col 2* </th>

</tr>

<tr>

<td> *Table Definition here* </td>

<td> …………………….. </td>

</tr>

</table>

1. Comments:

< ! - - *Your Comments here* - - >

1. Background Images:

<body background = “*pathname*/abc.gif”>

…………………….

</body>

Before we move on further, we need to know how the web exactly works.

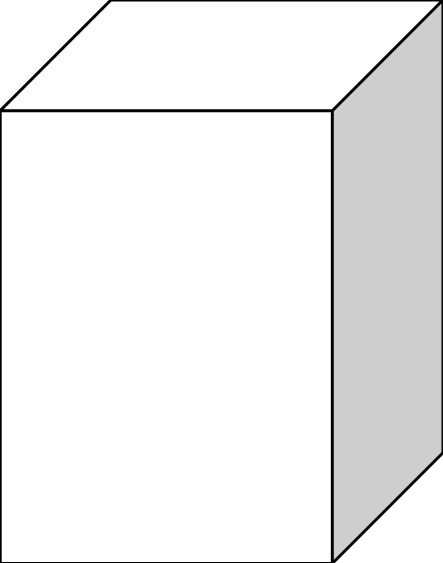
How does the Web work?

The Web is usually accessed through a browser. When the user types in a URL say, [www.revainstitution.org](http://www.heythere.com/) in the Address Bar of the browser, the browser makes a socket (Network) connection to the server [www.revainstitution.org](http://www.heythere.com/). This name is mapped to an IP address which is of the form 1.2.3.4 by making use of a DNS Server. The browser connects to this server using a logical *port 80*, the port that the server OS opens for internet connections. This port is standardized.

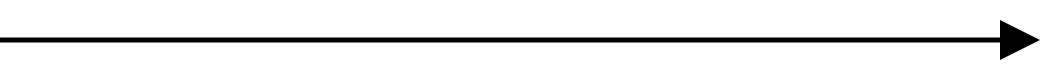
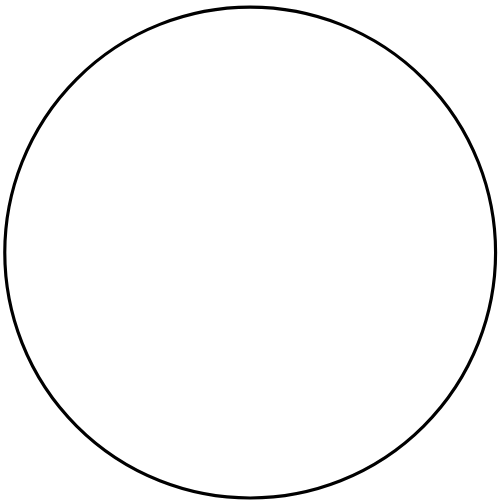
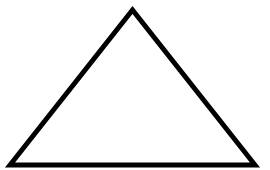
Based on the client request, the server delivers information. The type of data that the server sends back to the client could be a simple plain text (HTML), images, Java Applets (More about Applets later!) etc. this data can be obtained and delivered in three ways.

* *Serving Static Data* – The server does not do any kind of the processing. It merely obtains the data present on its local hard disk and sends it back to the client.
* *Serving Dynamic Data* – The Server does some processing in this case like executing a program and then outputs the result of the program back to the client as a response.
* *Serving Content with Embedded HTML* – Here, an executable code is present with the HTML file. It’s not quite static or dynamic.

Hyper links to heythere.com

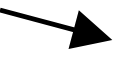


Browser



Disk

abcd



Web Server

W

*The Client: -*

*The Server: -*

**The Internet**

TCP Connection

**Basics of MySQL: -**

MySQL is an Open source Standard Query Language (SQL) database that is fast, reliable, easy to use and suitable for applications of any size. MySQL can be integrated into Perl programs by using the Perl *DBI (Database Independent Interface)* module. DBI is an API that allows Perl to connect and query a number of SQL Databases such as MYSQL, Oracle, Sybase etc.

For some of the programs in the Lab course, the MySQL database is to be used. For that, the MySQL Server is to be started. The following steps are to be performed in the same sequence on the Linux shell to start the server and create the database along with the table.

**To Start MySQL Server:**

**# mysql**

**mysql> create database reva; mysql> show databases; mysql> use reva; mysql> create table employee ( name varchar(25), age int );**

**mysql> insert into employee values (“e1”,21); mysql> insert into employee values (“e2”,22);**

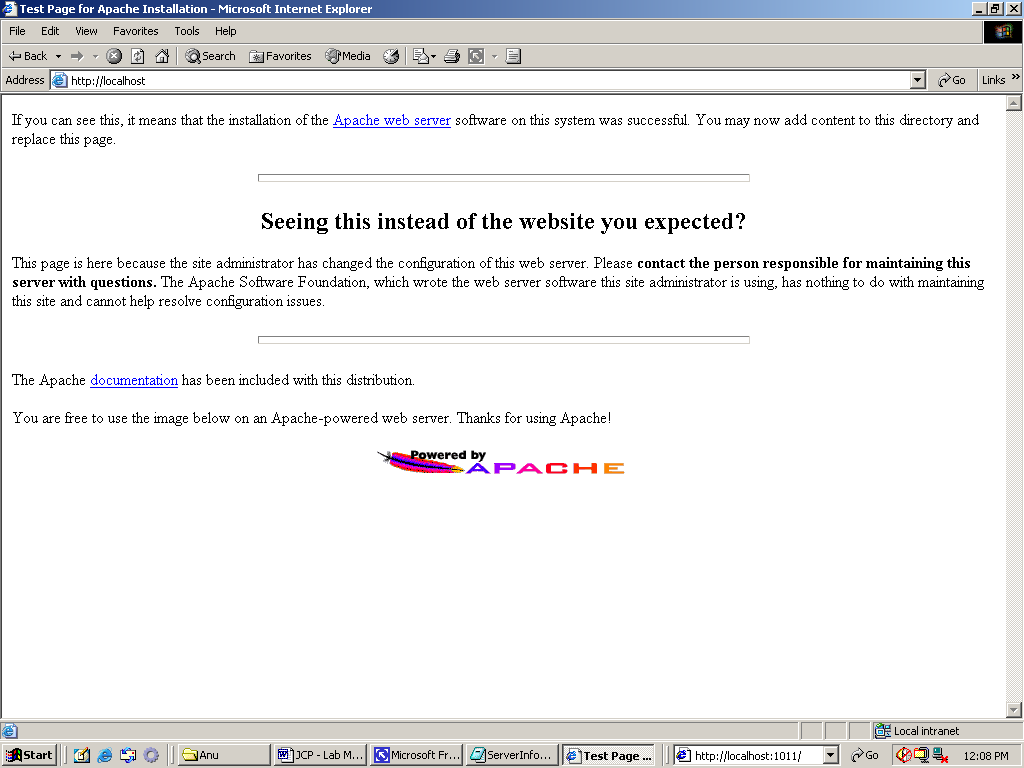
**mysql> exit;**

#.....*(The MySQL server is now started and a database along with a table called “employee” is ready for use).*

Let us examine each of these steps in detail now.

Apache Http Server:

The web server we are using here is Apache Http Server. It is freely downloadable from the site [www.apache.org](http://www.apache.org/). Once you have downloaded the installer, double click on that and install it in to your system. Then go to start menu  programs Apache HTTP Server Control Apache Server. Then click on start to start your server. Then open an Internet explorer and type <http://localhost:80/>. The port number 80 is *optional* in the URL. If you have successfully installed the server then you will get a screen shown below:



**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **SL. NO** | **TITLE OF THE PROGRAM** | **PAGE NUMBER** |
| **1** | **In today’s digital world, information dissemination through printed documents consume lot of time. To overcome this drawback it is better to adopt digital technology for information dissemination, like e- journals, e-books, e-advertisements, etc. Information dissemination through Internet in the form of web content is essential and convenient option. Design and develop static web pages for an online Book store. The pages should resemble like Error! Hyperlink reference not valid. Website should consist of Homepage, Registration & Login, User profile page, Books catalog, Shopping cart, Payment by credit card, and order confirmation.** | **8** |
| **2** | **Write an HTML page that has one input, which can take multi line text and a submit button. Once the user clicks the submit button, it should show the number of characters, words and lines in the text entered using an alert message. Words are separated with white space and lines are separated with new line character.** | **12** |
| **3** | **Internet or online services works on clients and server model. A client is a web browser through which users make requests, which contain input required, for service from the server to perform tasks. Server is a program running on a dedicated computer. Performance of any service or server depends on its throughput. Server throughput deteriorates when users send more and more invalid requests for service and thus results in wastage of server resources that are very precious. As a solution to this problem design a web page that takes student details such as Name, Semester, SRN, date of admission, email id and check for validity or correctness of the input data by writing a JavaScript to validate these fields.** | **14** |
| **4** | **Most of the organizations are conducting online tests for the placements. Similarly, online courses are conducting tests immediately after the course online to display the score instantly. Create an html page to show online exam using JavaScript. Let the page displays four Questions and have four optional answers using radio buttons and by clicking submit button display the score in an alert box. Reset the page into initial condition for next exam.** | **17** |
| **5** | **Develop and demonstrate JavaScript with POP-UP boxes and functions for the following problems:**   1. **Input: Click on Display Date button using onclick( ) function**   **Output: Display date in the textbox**   1. **Input: A number n obtained using prompt**   **Output: Factorial of n number using alert**   1. **Input: A number n obtained using prompt**   **Output: A multiplication table of numbers from 1 to 10 of n using alert**   1. **Input: A number n obtained using prompt and add another number using confirm**   **Output: Sum of the entire n numbers using alert.** | **20** |
| **6** | **PHP is a server scripting language, tool for making and powerful dynamic and interactive Web pages. Write a PHP program to store current date-time in a COOKIE and display the Last visited on date-time.** | **26** |
| **7** | **PHP (recursive acronym for PHP: Hypertext Preprocessor) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML. Write a PHP program to store page views count in SESSION, to increment the count on each refresh, and to show the count on web page.** | **27** |
| **8** | **In any business organization, employees keep traveling across different geographical locations and at the same time they want to be connected to server, file server, etc. to retrieve information such as sales details, assigning tasks to employees, and upload inspection site details, so on. Using PHP develop a web page that accepts book information such as ISBN number, title, authors, edition and publisher and store information submitted through web page in MySQL database. Design another web page to search for a book based on book title specified by the user and displays the search results with proper headings.** | **28** |
| **VIVA QUESTIONS** | | **31** |

**Program No: 1**

In today’s digital world, information dissemination through printed documents consume lot of time. To overcome this drawback it is better to adopt digital technology for information dissemination, like e- journals, e-books, e-advertisements, etc. Information dissemination through Internet in the form of web content is essential and convenient option. Design and develop static web pages for an online Book store. The pages should resemble like Error for Hyperlink reference not valid. Website should consist of Homepage, Registration & Login, User profile page, Books catalog, Shopping cart, Payment by credit card, and order confirmation.

**Algorithm**

**Step 1:** Include the standards of XHTML document

**Step 2:** Create various static web page.

**Step 3:** Use appropriate tag to link pages from one to other.

**Main.html**

<html>

<head>

<title>Main Page</title>

<style type="text/css">

a

{

color:Black;

font-size:large;

}

</style>

</head>

<body style="background-image: url(BgImg2.jpg); background-repeat: no-repeat; background-size: 100%;">

<center>

<h1 style="color:White;">

Main Page</h1>

<br />

<p>

<b><a href="Home.html">Home</a></b>

</p>

<p>

<b><a href="SignUp.html">Register</a></b>

</p>

<p>

<b><a href="Order.html">Order</a></b>

</p>

</center>

</body>

</html>

**Home.html**

<html>

<head>

<title>Home</title>

</head>

<body style="background-image: url(cbgimg1.jpg); background-repeat: no-repeat; background-size: 100%; color:White;">

<center>

<h1>Welcome to e-Books</h1>

</center>

<p>Select your book</p>

<hr />

<center>

<p>

<a href="CheckOut.html"><img src="BgImg1.jpg" height="250" width="250" /></a>

<a href="CheckOut.html"><img src="BgImg1.jpg" height="250" width="250" /></a>

<a href="CheckOut.html"><img src="BgImg1.jpg" height="250" width="250" /></a>

<a href="CheckOut.html"><img src="BgImg1.jpg" height="250" width="250" /></a>

<a href="CheckOut.html"><img src="BgImg1.jpg" height="250" width="250" /></a>

</p>

</center>

<center><a href="MainPage.html">Go to Main page</a></center>

</body>

</html>

**Checkout.html**

<html>

<head>

<title>Checkout</title>

</head>

<body style="background-image: url(cbgimg1.jpg); background-repeat: no-repeat; background-size: 100%; color:White;">

<center><h1>Checkout</h1></center>

<p>Enter Card details</p>

<hr />

Card No : <input type="text" /> <br /><br />

Name on Card : <input type="text" /> <br /><br />

Expiry date : <select >

<option>2018</option>

<option>2019</option>

<option>2020</option>

<option>2021</option>

</select> <br /><br />

CVV No : <input type="text" /> <br /><br />

Amount paid : <input type="text" /> <br /><br />

<input type="submit" value="Submit" /><input type="reset" value="reset" /> <br />

<center>

<a href="MainPage.html">Go to Main Page</a>

</center>

</body>

</html>

**Order.html**

<html>

<head>

<title>Order</title>

</head>

<body style="background-image: url(cbgimg1.jpg); background-repeat: no-repeat; background-size: 100%; color:White;">

<center>

<h1>

Order</h1>

<hr />

<p>

Your Order</p>

<a href="MainPage.html">Go to Main Page</a></center>

</body>

</html>

**Signup.html**

<html>

<head>

<title>Sign up</title>

</head>

<body style="background-image: url(cbgimg1.jpg); background-repeat: no-repeat; background-size: 100%; color:White;">

<center><h1>Sign up</h1></center>

<p>Enter your details</p>

<hr />

First Name : <input type="text" /> <br /><br />

Last Name : <input type="text" /> <br /><br />

Email Id : <input type="text" /> <br /><br />

User Id : <input type="text" /> <br /><br />

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

Phone No : <input type="text" /> <br /><br />

Day : <input type="text" /> <br /><br />

<input type="submit" value="Submit" /><input type="reset" value="reset" /> <br />

<center>

<a href="MainPage.html">Go to Main Page</a>

</center>

</body>

</html>

**OUTPUT:**

Main.html



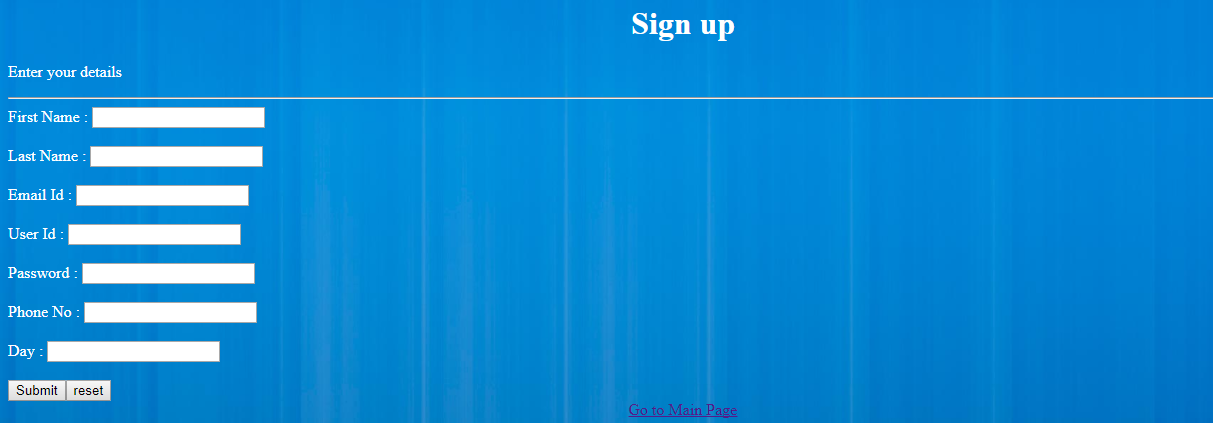
Home



Select a book



Register



**Program No: 2**

Write an HTML page that has one input, which can take multi line text and a submit button. Once the user clicks the submit button, it should show the number of characters, words and lines in the text entered using an alert message. Words are separated with white space and lines are separated with new line character.

**Algorithm**

**Step 1:** Create a html file.

**Step 2:** make use of textarea tag to collect input.

**Step 3:** write script to find out the no of words, characters and line in given input.

**Step 4:** display the output using alert method.

**2.html**

<html>

<head>

<script type="text/javascript">

function countWCL() {

var textarea=document.getElementById("tarea");

var text = textarea.value;

value = "Words: " + (text.split(/\b\S+\b/).length - 1) + " Characters: " + text.replace(/\s/g, "").length + "/" + text.replace(/\n/g, "").length + "lines:" + text.split("\n").length;

alert(value); }

</script></head>

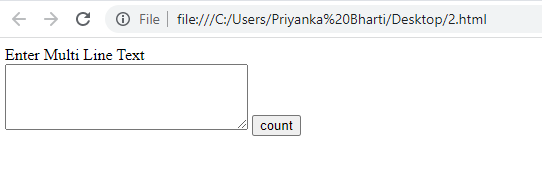
<form name="cwl">

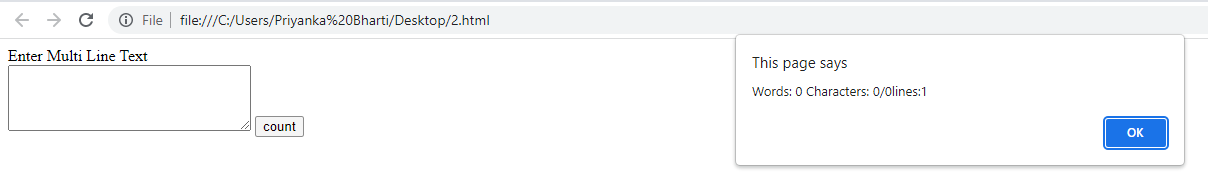
Enter Multi Line Text <br>

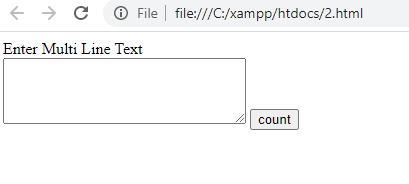
<textarea name="string" id="tarea" rows=4 cols=30></textarea>

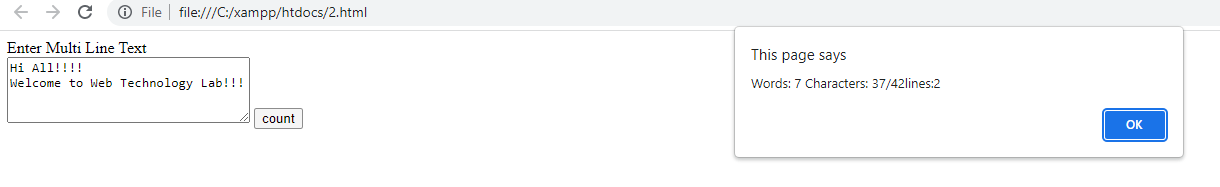
<input type="button" name="sub" value="count" onClick="countWCL()">

</form> </html>









**Program No: 3**

Internet or online services works on clients and server model. A client is a web browser through which users make requests, which contain input required, for service from the server to perform tasks. Server is a program running on a dedicated computer. Performance of any service or server depends on its throughput. Server throughput deteriorates when users send more and more invalid requests for service and thus results in wastage of server resources that are very precious. As a solution to this problem design a web page that takes student details such as Name, Semester, SRN, date of admission, email id and check for validity or correctness of the input data by writing a JavaScript to validate these fields.

**3.html**

<html>

<head>

<title>Student Registration</title>

</head>

<body>

<div align="Left">

<h1>Student Registration Portal</h1>

<form id="xyz">

<label for="name">Enter name: </label>

<input type="text" id="name" /><br /><hr />

<label for="dob">Select date of birth: </label>

<input type="date" id="dob" /><br /><hr />

<label for="branch">Enter branch: </label>

<input type="text" id="branch" /><br /><hr />

<label for="semester">Select Semester: </label>

<input type="number" id="semester" max="8" min="0" /><br /><hr />

label for="doj">Select date of joining: </label>

<<input type="date" id="doj" /><br /><hr />

<label for="university">Enter University Name: </label>

<input type="text" id="university" /><br /><hr />

<label for="mobile">Enter mobile number: </label>

<input type="text" id="mobile" /><br /><hr />

<label for="email\_add">Enter email: </label>

<input type="email" id="email\_add" /><br /><hr />

</form>

<button onclick="validate()">Submit</button>

<p> Result: <span id="result"></span> </p>

</div>

</body>

<script>

function validate()

{

var result\_text = document.getElementById("result");

var dob = document.getElementById("dob").value;

var birth\_year = parseInt(dob.substring(0, 4));

var doj = document.getElementById("doj").value;

var join\_year = parseInt(doj.substring(0,4));

if (join\_year - birth\_year < 17)

{

result\_text.innerHTML = "Too young to have started college!"

return;

}

var branch = document.getElementById("branch").value;

if (branch.search(/(CSE|ECE|ME|CE|EEE|BCA|MCA)/i) == -1)

{

result\_text.innerHTML = "Invalid branch!";

return;

}

var mobile\_no = document.getElementById("mobile").value;

if (mobile\_no.search(/^[0-9]+$/) == -1 || mobile\_no.length != 10)

{

result\_text.innerHTML = "Invalid mobile number!";

return;

}

var email = document.getElementById("email\_add").value;

if (email.search(/^(([^<>()\[\]\\.,;:\s@"]+(\.[^<>()\[\]\\.,;:\s@"]+)\*)|(".+"))@((\[[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}])|(([a-zA-Z\-0-9]+\.)+[a-zA-Z]{2,}))$/) == -1)

{

result\_text.innerHTML = "Invalid email ID";

return;

}

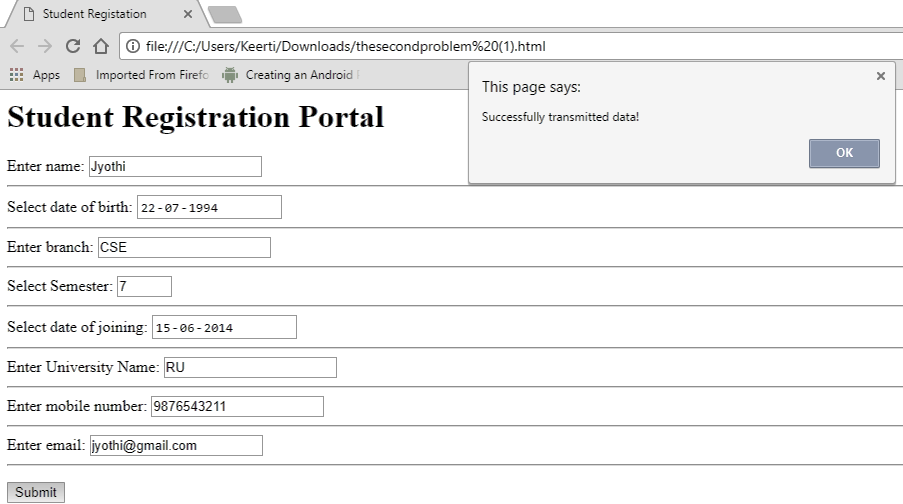
alert('Successfully transmitted data!');

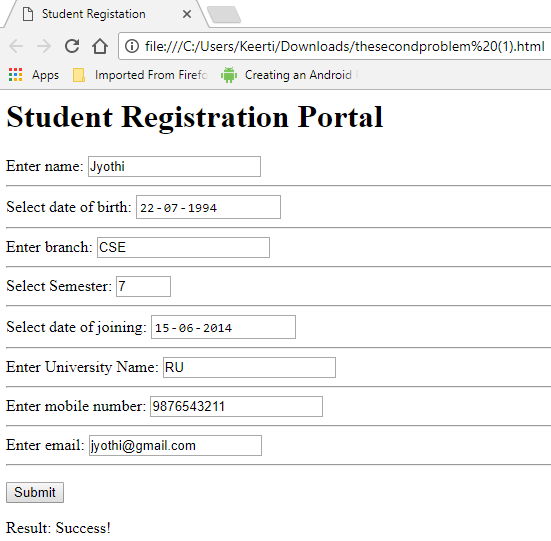
result\_text.innerHTML = "Success!";

}

</script>

</html>





**Program 4**

Most of the organizations are conducting online tests for the placements. Similarly, online courses are conducting tests immediately after the course online to display the score instantly. Create an html page to show online exam using JavaScript. Let the page displays four Questions and have four optional answers using radio buttons and by clicking submit button display the score in an alert box. Reset the page into initial condition for next exam.

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<style>

div#test{ border:#000 1px solid; padding:10px 40px 40px 40px; }

</style>

<script>

var pos = 0, test, test\_status, question, choice, choices, chA, chB, chC, correct = 0;

var questions = [

[ "What is 10 + 4?", "12", "14", "16", "B" ],

[ "What is 20 - 9?", "7", "13", "11", "C" ],

[ "What is 7 x 3?", "21", "24", "25", "A" ],

[ "What is 8 / 2?", "10", "2", "4", "C" ]

];

function \_(x){

return document.getElementById(x);

}

function renderQuestion(){

test = \_("test");

if(pos >= questions.length){

test.innerHTML = "<h2>You got "+correct+" of "+questions.length+" questions correct</h2>";

\_("test\_status").innerHTML = "Test Completed";

pos = 0;

correct = 0;

return false;

}

\_("test\_status").innerHTML = "Question "+(pos+1)+" of "+questions.length;

question = questions[pos][0];

chA = questions[pos][1];

chB = questions[pos][2];

chC = questions[pos][3];

test.innerHTML = "<h3>"+question+"</h3>";

test.innerHTML += "<input type='radio' name='choices' value='A'> "+chA+"<br>";

test.innerHTML += "<input type='radio' name='choices' value='B'> "+chB+"<br>";

test.innerHTML += "<input type='radio' name='choices' value='C'> "+chC+"<br><br>";

test.innerHTML += "<button onclick='checkAnswer()'>Submit Answer</button>";

}

function checkAnswer(){

choices = document.getElementsByName("choices");

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

if(choices[i].checked){

choice = choices[i].value;

}

}

if(choice == questions[pos][4]){

correct++;

}

pos++;

renderQuestion();

}

window.addEventListener("load", renderQuestion, false);

</script>

</head>

<body>

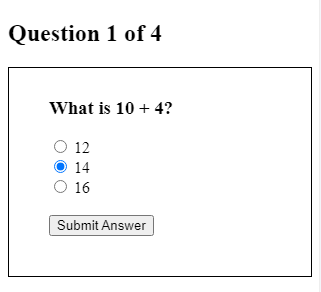
<h2 id="test\_status"></h2>

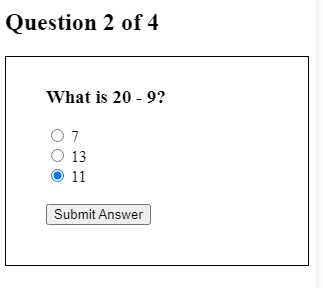
<div id="test"></div>

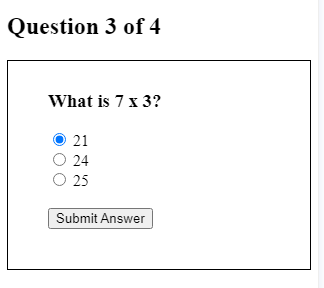
</body>

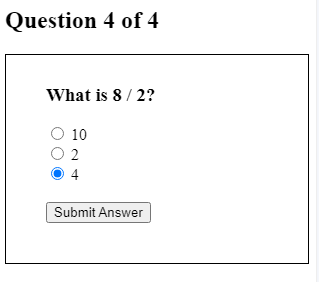
</html>

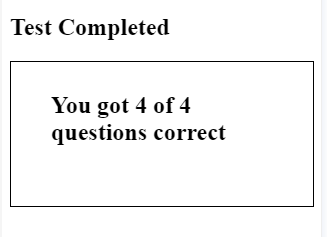
**Output:**











**5. Develop and demonstrate JavaScript with POP-UP boxes and functions for the**

**Following problems:**

**a) Input:** Click on Display Date button using onclick( ) function Output: Display date in the

textbox

**b) Input:** A number n obtained using prompt Output: Factorial of n number using alert

**c) Input:** A number n obtained using prompt Output: A multiplication table of numbers from 1

to 10 of n using alert

**d)Input:** A number n obtained using prompt and add another number using confirm

**Output:** Sum of the entire n numbers using alert

<html>

<body>

<title>date</title>

<script>

function display(){

var x="You have clicked";

var d=new Date();

var date=d.getDate();

var month=d.getMonth();

month++;

var year=d.getFullYear();

document.getElementById("dis").value=date+"/"+month+"/"+year;

}

</script>

<form>

<input type="text" id="dis" /><br />

<input type="button" value="Display Date" onclick="display()" />

</form>

<body>

</html>

Graphical user interface, application, Word

Description automatically generated

Graphical user interface, application, Word

Description automatically generated

<html>

<head>

<title>Factorial</title>

<script type="text/javascript">

function factorialcalc()

{

var number = prompt("Enter a number" ) ;

var factorial = 1 ;

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

{

factorial = factorial \* i ;

}

alert("The factorial of " + number + " is " + factorial) ;

}

</script>

</head>

<body><form name=frm>

<input type=button value='factorial' onclick="factorialcalc();">

</form>

</body>

</html>

Graphical user interface, application, Word

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

<html>

<head><title> Multiplication Table </title></head>

<body>

<script type="text/javascript">

var n=prompt("Enter positive value for n: "," ");

if(!isNaN(n)) {

var table="";

var number="";

for(i=1;i<=10;i++) {

number = n \* i;

table += n + " \* " + i + " = " + number + "\n";

}

alert(table);

}

else {

alert("Enter positive value");

n=prompt("Enter positive value for n: "," ");

}

document.write(n+" table values displayed using alert ..<br />");

</script>

</body>

</html>

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application, Word

Description automatically generated

<html>

<head><title>sum of n numbers using popup boxes</title>

<script type="text/javascript">

function addsum()

{

alert("you're going to give me a list of numbers. i'm going to add them together for you");

var keepgoing = true;

var sumofnums = 0;

while (keepgoing) {

sumofnums = sumofnums + (parseInt(prompt("what's the next number to add?",""))) ;

keepgoing = confirm("add another number?") ;

}

alert("the sum of all your numbers is " + sumofnums) ;

}

</script>

</head>

<body>

<form name=frm>

<input type=button value='sum of n numbers' onclick="addsum();">

</form>

</body>

</html>

Graphical user interface, application, Word

Description automatically generated

Graphical user interface, application, Word

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

**Program No: 6**

PHP is a server scripting language, tool for making and powerful dynamic and interactive Web pages. Write a PHP program to store current date-time in a COOKIE and display the Last visited on date-time.

**6.html**

<html >

<head> <title>Cookies</title> </head>

<body>

<form action= “6.php" method="post">

<p> The last visited time was <input type="submit" value="Display Now"/> </p>

</form>

</body>

</html>

**6.php**

<?php

$present\_time=date(“H:i:s-m/d/y”);

$expiry= 60 \* 60 \*24 \*60 + time();

setcookie("Lastvisit",$present\_time, $expiry);

if(isset($\_COOKIE[“Lastvisit”]))

{

echo "Cookie has been set";

echo “The current time of the system is”;

echo $present\_time;

echo "The Last visited Time and Date";

echo $\_COOKIE["Lastvisit"];

}

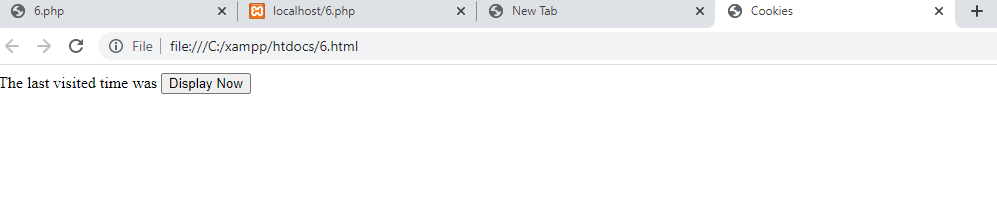
else

echo ”You’ve got some old cookies!”;

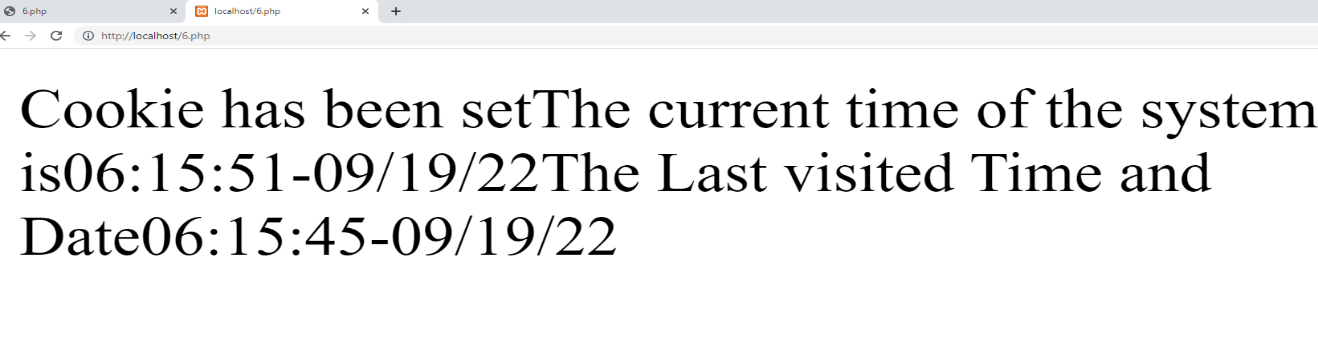
?>

**OUTPUT:**

**6.html**



**6.php**



**Program No: 7**

PHP (recursive acronym for PHP: Hypertext Preprocessor) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML. Write a PHP program to store page views count in SESSION, to increment the count on each refresh, and to show the count on web page.

**7.html**

<html>

<head> <title>SESSION PROGRAM </title> </head>

<body>

<form action=" 7.php" method="post">

<p> To see page views count in session <input type="submit" value="Click Here"/> </p>

</form>

</body>

</html>

**7.php**

<?php

session\_start();

if (!isset($\_SESSION))

{

$\_SESSION["count"] = 1;

echo "<p>Counter initialized</p>\n";

}

else { $\_SESSION["count"]++; }

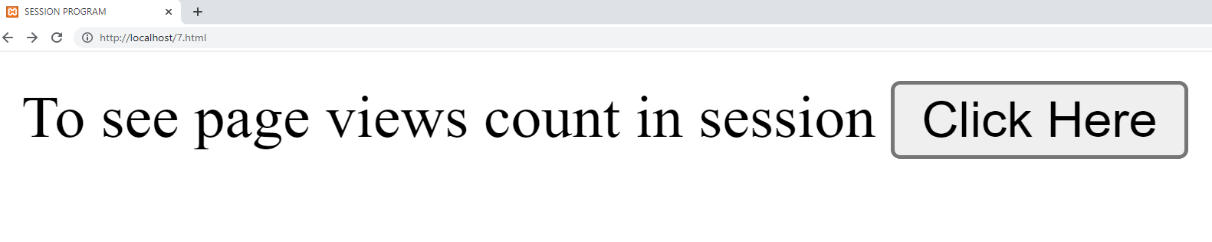
echo "<p>This page has been viewed <b>$\_SESSION[count]</b> times.</p>".

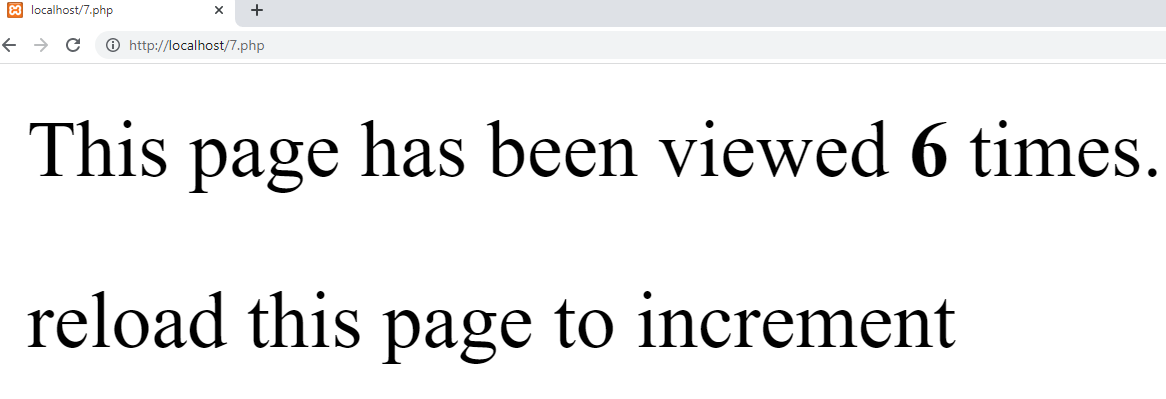
"<p>reload this page to increment</p>";

?>

**OUTPUT:**

**7.html**

****

****

**Program No: 8**

In any business organization, employees keep traveling across different geographical locations and at the same time they want to be connected to server, file server, etc. to retrieve information such as sales details, assigning tasks to employees, and upload inspection site details, so on. Using PHP develop a web page that accepts book information such as ISBN number, title, authors, edition and publisher and store information submitted through web page in MySQL database. Design another web page to search for a book based on book title specified by the user and displays the search results with proper headings.

First Name Last Name Age Address Pincode

Ram Kumar 21 REVA University 560064

Anil Vinay 30 REVA University 560064

**Step 1:** Create three separate files. One of type XHTML and two of PHP.

**Step 2:** In XHTML doc, create text box to ISBN Number, title, authors, edition and publisher. Create submit button and send this as a input to a insert php file.

**Step 3:** In the insert php file, make the connection to a database.

**Step 4:** Once the connection to a database is made successfully, a particular database is selected and all the values are entered in a particular table by writing the following query:

$sql="INSERT INTO Book (ISBN, title, authors, edition, publisher)

VALUES ('$\_POST [ISBN]','$\_POST [title]','$\_POST [authors]','$\_POST [edition]', '$\_POST [publisher])";

**Step 5:** In the insert php file, a text box is created to take the book name based on which the record has to be searched.

**Step 6:** The name is sent as an input to a result php file .

**Step 7:** In the result php file , the connection to the database is made and table is selected.

**Step 8:** Then all the records are searched corresponding the entered name .

**Step 9:** The connection to the database is terminated .

**Program8.html**

<html>

<body>

<form action="program8insert.php" method="post">

ISBN: <input type="text" name="ISBN" />

Title: <input type="text" name="Title" />

Authors: <input type="text" name="authors" />

Edition: <input type="text" name="edition" />

Publisher: <input type="text" name="publisherl" />

<input type="submit" />

</form>

</body>

</html>

**Program 8insert.php**

<html>

<body>

<?php

$con = mysql\_connect("localhost","root","");

if (!$con)

{

die('Could not connect: ' . mysql\_error());

}

mysql\_select\_db("cse");

$sql="INSERT INTO Book (ISBN,title, authors, edition,publisher)

VALUES ('$\_POST[ISBN]','$\_POST[Title]','$\_POST[Author]','$\_POST[Edition]',’$\_POST[Publisher]’)";

if (!mysql\_query($sql,$con))

{

die('Error: ' . mysql\_error());

}

echo "1 record added";

mysql\_close($con)

?>

<form action="program8results.php" method="post">

Name: <input type="text" name="Title" />

<input type="submit" />

</form>

</body>

</html>

**program8results.php**

<html>

<body>

<?php

$con = mysql\_connect("localhost","root","");

if (!$con)

{

die('Could not connect: ' . mysql\_error());

}

mysql\_select\_db("cse");

$result = mysql\_query("SELECT \* FROM Book where name= '$\_POST[Title]'");

if(!$result)

{

echo "There is no records";

}

echo "<table border='1'>

<tr>

<th>ISBN</th>

<th>Title</th>

<th>Authors</th>

<th>Edition </th>

<th>Publisher </th>

</tr>";

while($row = mysql\_fetch\_array($result))

{

echo "<tr>";

echo "<td>" . $row['ISBN'] . "</td>";

echo "<td>" . $row['Title'] . "</td>";

echo "<td>" . $row['Authors'] . "</td>";

echo "<td>" . $row['Edition'] . "</td>";

echo "<td>" . $row['Publisher'] . "</td>";

echo "</tr>";

}

echo "</table>";

mysql\_close($con);

?>

</body>

</html>

**Viva Questions**

1. Why HTML is called a “markup” language?
2. Can you name some Markup languages other than HTML?
3. How do you write comments in HTML?
4. What does the <br> and the <p> tag in HTML do?
5. How do you insert bullets in a HTML page?
6. What is the difference between Dynamic HTML and Dynamic Web Pages?
7. What are the Components of URL?
8. What does ‘80’ mean in localhost: 8080?
9. What is the difference between system( ) and backticks(`)?
10. Why the “\n\n” is used in print “Content-type:text/html\n\n”;? Will the program execute even without it?
11. What does the IP address 127.0.0.1 correspond to?
12. What is DBI? What is “independent” about it?
13. What does “standard” do? What is the alternative to “standard”?
14. What are cookies? Why do you need them?
15. What do you mean by a “Session”? Comment on its importance.
16. What do you mean by URI? Is it the same as URL?
17. What is a Query String?
18. What do you mean by “Embedded HTML”?
19. What does the “-w” option in the Interpreter line in Perl Programs do?
20. How do you declare functions in Perl?
21. How do you write comments in Perl?
22. What does the “short-circuit” logic operator used in languages like Perl, Java do?
23. Why do you have to “use” a database that you have created in MySQL before you perform other operations?
24. What are the different data types supported by MySQL?
25. What do you mean by a BLOB?
26. What does the “grant” command in MySQL do?
27. What does the die ( ) do?
28. What do you mean by the “Path Information”?
29. What is the difference between “GET” and “POST”?
30. What is the difference between doGet ( ) and doPost ( ) methods used in Servlet