

GUDLAVALLERU ENGINEERING COLLEGE

(An Autonomous Institute with Permanent Affiliation to JNTUK, Kakinada)
Seshadri Rao Knowledge Village, Gudlavalleru – 521 356.

Department of Computer Science and Engineering

III Year I Semester

2019 - 2020



WEB TECHNOLOGIES LAB FACULTY MANUAL

Prepared by

Dr. K. Bala Brahmewara
Assistant Professor

Mr. K. Hareesh Kumar
Assistant Professor

Mrs. T. Nagamani
Assistant Professor

Mrs. Y. Aditya
Assistant Professor

GUDLAVALLERU ENGINEERING COLLEGE

**(An Autonomous Institute with Permanent Affiliation to JNTUK, Kakinada)
Seshadri Rao Knowledge Village, Gudlavalleru – 521 356.**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE VISION & MISSION

Institute Vision:

To be a leading institution of engineering education and research, preparing students for leadership in their fields in a caring and challenging learning environment.

Institute Mission:

- To produce quality engineers by providing state-of-the-art engineering education.
- To attract and retain knowledgeable, creative, motivated and highly skilled individuals whose leadership and contributions uphold the college tenets of education, creativity, research and responsible public service.
- To develop faculty and resources to impart and disseminate knowledge and information to students and also to society that will enhance educational level, which in turn, will contribute to social and economic betterment of society.
- To provide an environment that values and encourages knowledge acquisition and academic freedom, making this a preferred institution for knowledge seekers.
- To provide quality assurance.
- To partner and collaborate with industry, government, and R&D institutes to develop new knowledge and sustainable technologies and serve as an engine for facilitating the nation's economic development.
- To impart personality development skills to students that will help them to succeed and lead.
- To instil in students the attitude, values and vision that will prepare them to lead lives of personal integrity and civic responsibility.
- To promote a campus environment that welcomes and makes students of all races, cultures and civilizations feel at home.
- Putting students face to face with industrial, governmental and societal challenges.

DEPARTMENT VISION & MISSION

VISION

To be a Center of Excellence in computer science and engineering education and training to meet the challenging needs of the industry and society.

MISSION

- To impart quality education through well-designed curriculum in tune with the growing software need of the industry.
- To serve our students by inculcating in them problem solving, leadership, teamwork skills and the values of commitment to quality, ethical behavior & respect for other.
- To foster industry-academia relationship for mutual benefit and growth.

PROGRAMME EDUCATIONAL OBJECTIVES(PEOs):-

PEO1: Identify, analyze, formulate and solve Computer Science and Engineering problems both independently and in a team environment by using the appropriate modern tools.

PEO2: Manage software projects with significant technical, legal, ethical, social, environmental and economic considerations.

PEO3: Demonstrate commitment and progress in lifelong learning, professional development, leadership and Communicate effectively with professional clients and the public.

PROGRAM OUTCOMES (POs)

Engineering students 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.

PROGRAM SPECIFIC OUTCOMES

Students will be able to

PSO1: Organize, maintain and protect IT Infrastructural resources.

PSO2: Design and Develop web, mobile, and smart apps based software solutions to the real world Problems.

WEB TECHNOLOGIES LAB

Course Objectives

- To introduce concepts of designing dynamic web pages using HTML, CSS and Javascript.
- To familiarize with JSP programming and master database access using JSP and JDBC.
- To impart PHP programming and master database access using PHP and MySQL.

Course Outcomes

Upon successful completion of the course, the students will be able to

- design dynamic web pages using HTML, CSS and JavaScript.
- access and Validate form data using JavaScript and PHP.
- connect to database using JSP and JDBC and perform various operations.
- connect to MySQL using PHP and perform various operations

Mapping of course outcomes with program outcomes

CT2520 : WEB TECHNOLOGIES LAB														
Course outcomes	Program Outcomes and Program Specific Outcome													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1: design dynamic web pages using HTML, CSS and JavaScript.	2	2	3					1	1	1		2	1	3
CO2: access and Validate form data using JavaScript and PHP.	1		2		2			2	1	1		2	1	2
CO3: connect to database using JSP and JDBC and perform various operations.	2	1	2		2			2	2	2		2	2	3
CO4: connect to MySQL using PHP and perform various operations.	2	1	2		2			2	2	2		2	2	3
WEB TECHNOLOGIES LAB	2	1	3		2			2	2	2		2	2	3

INDEX

S.No	List of experiments	Page No
1	Design the following static web pages required for an online book store web site. i. Home Page ii. Login Page iii. Catalogue Page	7-11
2	Design the following static web pages required for an online book store web site. i. Registration Page ii. Cart Page	12-15
3	Design a web page using CSS which includes the following: i. Use different font and text styles ii. Set a background image for both the page and single element on the page. iii. Define styles for links iv. Working with layers v. Adding a Customized cursor	16-23
4	i. Write a JavaScript to validate the fields of the login page. ii. Write a JavaScript to validate the fields of the Registration page	24-34
5	Write an XML file which will display the Book information which includes the following: Title of the book, Author Name, ISBN number, Publisher name, Edition and Price. Validate the above document using DTD and XML Schema.	35-38
6	i. Write a PHP program to validate the fields of the login page. ii. Write a PHP program to validate the fields of the Registration page	39-44
7	Write a JSP to connect to the database and extract data from the tables and display them to the user.	45-46
8	Design a JSP to insert the details of the users who register through the registration page and store the details in to the database.	47-49
9	Write a PHP program to connect to MySQL database which retrieves the data from the tables and display them to the user.	50-56
10	Write a PHP program to insert the details entered by the user in the Registration form into MySQL database.	57-58

EXP NO: 1

AIM: Design the following static web pages required for an online book store web site.

i. Home Page ii. Login Page iii. Catalogue Page

1) HOME PAGE:

The static home page must contain three **frames**.

Top frame : Logo and the college name and links to Home page, Login page, Registration page, Catalogue page and Cart page (the description of these pages will be given below).

Left frame : At least four links for navigation, which will display the catalogue of respective links. For e.g.: When you click the link “CSE” the catalogue for CSE Books should be displayed in the Right frame.

Right frame: The pages to the links in the left frame must be loaded here. Initially this page contains description of the web site.

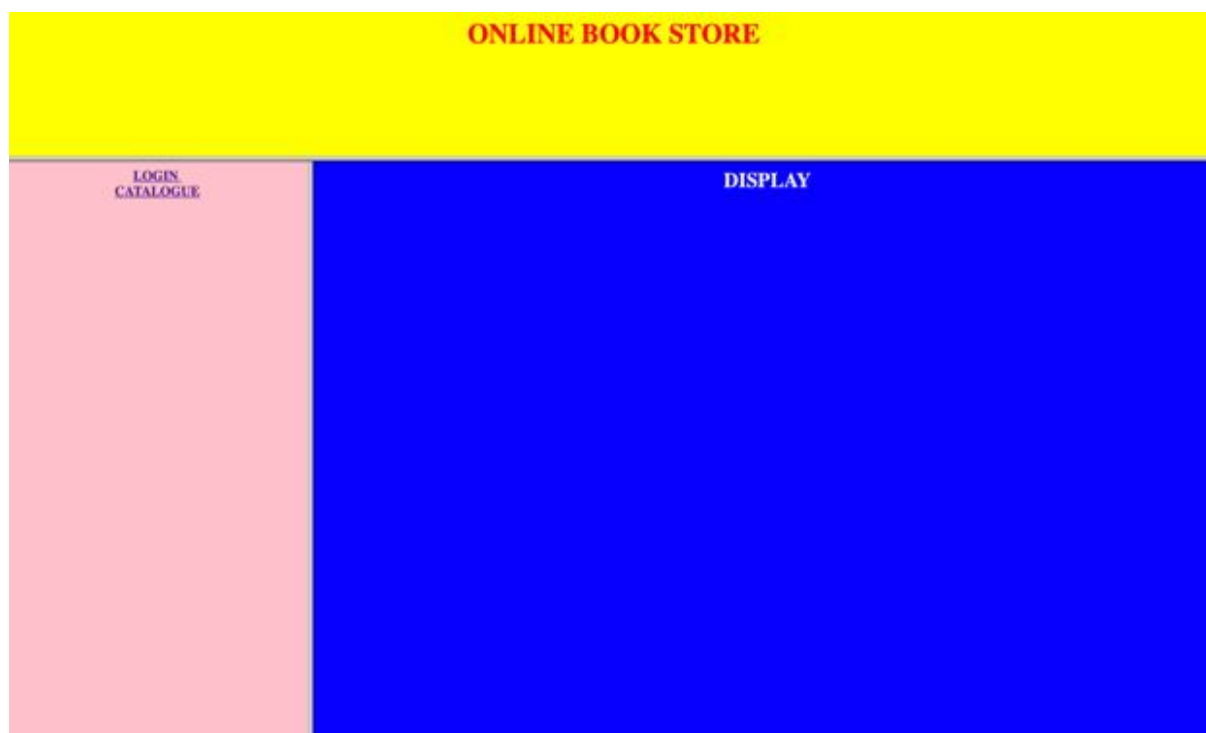


Fig 1.1

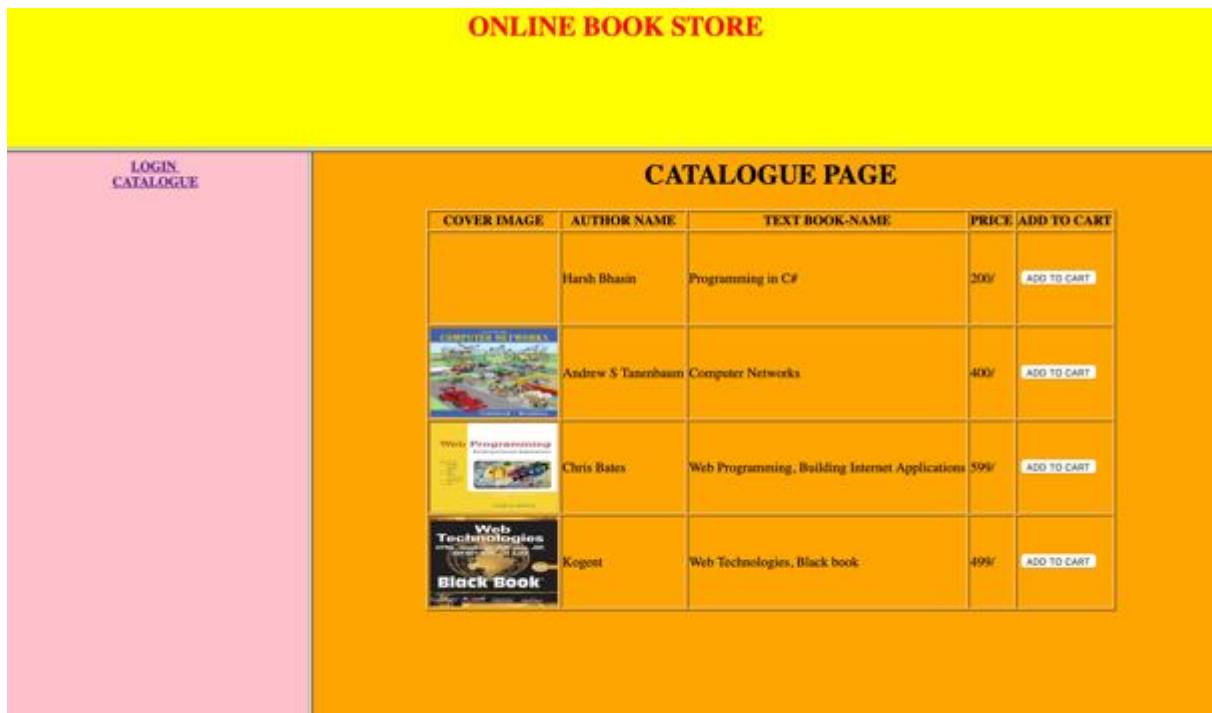
2) **LOGIN PAGE:** This page looks like below:



3) CATALOGUE PAGE:

The catalogue page should contain the details of all the books available in the web site in a table. The details should contain the following:

1. Snap shot of Cover Page.
2. Author Name.
3. Publisher.
4. Price.
5. Add to cart button.



Source Code

home.html

```
<html>
<head>
<frameset rows="20%,80%">
<frame name="title" src="titlepage.html">
<frameset cols="25%,75%">
<frame name="list" src="list.html">
<frame name="display" src="display.html">
</frameset>
</frameset>
</head>
</html>
```

titlepage.html

```
<html>
<head>
<title>title page</title>
</head>
<body bgcolor="yellow" text="red">
<h1 align="center"> ONLINE BOOK STORE </h1>
</body>
</html>
```

list.html

```
<html>
<head>
<title> List</title>
</head>
<body bgcolor="pink">
<center>
<a href="login.html" target="display"><b> LOGIN </b></a><br>
<a href="catalogue.html" target="display"><b> CATALOGUE </b></a>
</center>
</body>
</html>
```

Left.html

```
<html>
<head>
<title>Top Page</title>
</head>
<body>
<a href="cse.html" target="body">CSE</a><br />
<a href="ece.html" target="body">ECE</a><br />
<a href="eee.html" target="body">EEE</a><br />
<a href="civil.html" target="body">CIVIL</a><br />
```

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

Login.html

```
<html>
<body bgcolor="cyan">
<h2 align="center">LOGIN PAGE</h2>
<center>
<form>
<table border="1">
<tr>
<td><label>USER NAME</label></td>
<td><input type="text" name="uname"></td>
</tr>
<tr>
<td><label>PASSWORD</label></td>
<td><input type="password" name="pwd"></td>
</tr>
<tr>
<td><input type="submit" value="LOGIN"></td>
</tr>
</table>
</form>
</center>
</body>
</html>
```

Catalogue.html

```
<html>
<body bgcolor="orange" >
<h1 align="center">CATALOGUE PAGE</h1>
<center>
<table border="1">

<tr>
<th> COVER IMAGE</th>
<th> AUTHOR NAME</th>
<th> TEXT BOOK-NAME</th>
<th> PRICE</th>
<th> ADD TO CART</th>
</tr>
<tr>
<td><imgsrc="c#.jpg" height="105" width="150"></td>
<td>Harsh Bhasin</td>
<td> Programming in C#</td>
<td>200</td>
<td><input type="button" value="ADD TO CART"></td>
</tr>
<tr>
<td><imgsrc="cn.jpg" height="105" width="150"></td>
```

```

<td>Andrew S Tanenbaum</td>
<td>Computer Networks</td>
<td>400</td>
<td><input type="button" value="ADD TO CART"></td>
<tr>
<td><imgsrc="wt.jpg" height="105" width="150"></td>
<td>Chris Bates</td>
<td>Web Programming, Building Internet Applications</td>
<td>599</td>
<td><input type="button" value="ADD TO CART"></td>
<tr>
<td><imgsrc="wt1.jpg" height="105" width="150"></td>
<td>Kogent</td>
<td>Web Technologies, Black book</td>
<td>499</td>
<td><input type="button" value="ADD TO CART"></td>

</table>
</center>
</body>
</html>

```

display.html

```

<html>
<head>
<title>display</title>
</head>
<body bgcolor="blue" text="white">
<h2 align="center">DISPLAY</h2>
</body>
</html>

```

Execution Procedure

Step1: type all html programs in notepad or any text editor and save .html format.

Step2: open web browser and open index.html

Step3: traverse all pages from index.html.

EXP NO: 2

Aim: Design the following static web pages required for an online book store web site.

i. Registration Page ii. Cart Page

CART PAGE:

The cart page contains the details about the books which are added to the cart.

The cart page should look like this:



CART PAGE

Book Name	Price	Qty	Amount
Web Programming, Building Internet Applications	500	2	1000
Web Technologies, Black book	400	3	1200
Web Technologies	300	4	1200
Total Amount			3400

5) REGISTRATION PAGE:

Create a “registration form “with the following fields

CATALOGUE PAGE

COVER IMAGE	AUTHOR NAME	TEXT BOOK-NAME	PRICE	ADD TO CART
	Harsh Bhasin	Programing in C#	200/-	<input type="button" value="ADD TO CART"/>
	Andrew S Tanenbaum	Computer Networks	400/-	<input type="button" value="ADD TO CART"/>
	Chris Bates	Web Programming, Building Internet Applications	599/-	<input type="button" value="ADD TO CART"/>
	Kogent	Web Technologies, Black book	499/-	<input type="button" value="ADD TO CART"/>

Description

Through table and form tag we can design pages.

The form tag you can add to your web pages a guestbook, order forms, surveys, get feedback or whatever. The basic construction of a html form is this...

<FORM> begin a form

<INPUT> ask for information in one of several different ways...

<INPUT> ...there can be as many input areas as you wish

</FORM> end a form

<TABLE>

The main tag. Used to tell the browser "this is a table", along with some attributes like size, border width and a few other things.

<TR>

TableRow defines a horizontal row of **<TD>** (TableData) *cells*.

<TD>

Specifies an individual block or *cell* in a table row

Source Code

Registraion.html

```

<html>
<body bgcolor="seagreen">
<h1 align="center"><font color="red">CART PAGE</font></h1>
<center>
<table border="1">
<tr>
<th>Book Name</th>
<th>Price</th>
<th>Qty</th>
<th>Amount</th>
</tr>

```

```

<tr>
<td>Web Programming, Building Internet Applications</td>
<td>500/-</td>
<td>2</td>
<td>1000</td>
</tr>
<tr>
<td>Web Technologies, Black book</td>
<td>400/-</td>
<td>3</td>
<td>1200</td>
</tr>
<tr>
<td>Web Technologies</td>
<td>300/-</td>
<td>4</td>
<td>1200</td>
</tr>
<tr>
<td colspan="3">Total Amount</td>
<td>3400</td>
</tr>
</table>
</body>
</html>

```

Cart.html

```

<html>
<body bgcolor="orange" >
<h1 align="center">CATALOGUE PAGE</h1>
<center>
<table border="1">
<tr>
<th> COVER IMAGE</th>
<th> AUTHOR NAME</th>
<th> TEXT BOOK-NAME</th>
<th> PRICE</th>
<th> ADD TO CART</th>
</tr>
<tr>
<td><imgsrc="c#.jpg" height="105" width="150"></td>
<td>Harsh Bhasin</td>
<td> Programming in C#</td>
<td>200</td>
<td><input type="button" value="ADD TO CART"></td>
</tr>
<tr>
<td><imgsrc="cn.jpg" height="105" width="150"></td>
<td>Andrew S Tanenbaum</td>
<td>Computer Networks</td>

```

```
<td>400/</td>
<td><input type="button" value="ADD TO CART"></td>
<tr>
<td><imgsrc="wt.jpg" height="105" width="150"></td>
<td>Chris Bates</td>
<td>Web Programming, Building Internet Applications</td>
<td>599/</td>
<td><input type="button" value="ADD TO CART"></td>
<tr>
<td><imgsrc="wt1.jpg" height="105" width="150"></td>
<td>Kogent</td>
<td>Web Technologies, Black book</td>
<td>499/</td>
<td><input type="button" value="ADD TO CART"></td>
</table>
</center>
</body>
</html>
```

Execution Procedure

Step1: type all html programs in notepad or any text editor and save .html format.

Step2: open web browser and open index.html, i.e previous week.

Step3: traverse all pages from index.html.

EXP NO: 3

Design a web page using CSS which includes the following:

- i. Use different font and text styles
- ii. Set a background image for both the page and single element on the page.
- iii. Define styles for links
- iv. Working with layers
- v. Adding a Customized cursor

1) Use different font, styles:

In the style definition you define how each selector should work (font, color etc.). Then, in the body of your pages, you refer to these selectors to activate the styles.

For example:

```
<HTML>
<HEAD>
<style type="text/css">
B.headline {color:red, font-size:22px, font-family:arial, text-decoration:underline}
</style>
```

```
</HEAD>
```

```
<BODY>
```

```
<b>This is normal bold</b><br>
```

```
Selector {cursor:value}
```

For example:

```
<html>
<head>
<style type="text/css">
.xlink {cursor:crosshair}
.hlink {cursor:help}
</style>
</head>

<body>
<b>
<a href="mypage.htm" class="xlink">CROSS LINK</a>
<br>
<a href="mypage.htm" class="hlink">HELP LINK</a>
</b>
</body>
</html>

<b class="headline">This is headline style bold</b>
</BODY>

</HTML>
```


2) Set a background image for both the page and single elements on the page. You can define the background image for the page like this:

```
BODY {background-image:url(myimage.gif),}
```

3) Control the repetition of the image with the background-repeat property. As background-repeat: repeat
Tiles the image until the entire page is filled, just like an ordinary background image in plain HTML.

4) Define styles for links as

```
A:link
A:visited
A:active
A:hover
```

Example:

```
<style type="text/css">
A:link {text-decoration: none}
A:visited {text-decoration: none}
A:active {text-decoration: none}
A:hover {text-decoration: underline, color: red,}
</style>
```

5) Work with layers:

For example:

LAYER 1 ON TOP:

```
<div style="position:relative, font-size:50px, z-index:2,">LAYER 1</div><div
style="position:relative, top:-50, left:5, color:red, font-size:80px, z-
index:1">LAYER 2</div>
```

LAYER 2 ON TOP:

```
<div style="position:relative, font-size:50px, z-index:3,">LAYER 1</div><div
style="position:relative, top:-50, left:5, color:red, font-size:80px, z-
index:4">LAYER 2</div>
```

6) Add a customized cursor:

```
Selector {cursor:value}
```

For example:

```
<html>
<head>
<style type="text/css">
.xlink {cursor:crosshair}
.hlink {cursor:help}
</style>
</head>

<body>
<b>
<a href="mypage.htm" class="xlink">CROSS LINK</a>
<br>
```

```
<a href="mypage.htm" class="hlink">HELP LINK</a>
</b>
</body>
</html>
```

Description

- **CSS** stands for **Cascading Style Sheets**
- Styles define **how to display** HTML elements
- Styles are normally stored in **Style Sheets**
- Styles were added to HTML 4.0 **to solve a problem**
- **External Style Sheets** can save you a lot of work
- External Style Sheets are stored in **CSS files**
- Multiple style definitions will **cascade** into one

HTML tags were originally designed to define the content of a document.

As the two major browsers - Netscape and Internet Explorer - continued to add new HTML tags and attributes (like the tag and the color attribute) to the original HTML specification, it became more and more difficult to create Web sites where the content of HTML documents was clearly separated from the document's presentation layout.

To solve this problem, the World Wide Web Consortium (W3C) - the non profit, standard setting consortium, responsible for standardizing HTML - created STYLES in addition to HTML 4.0.

Source Code

1) Use different font, styles:

Fontstyle.html

```
<html>
<head>
<link rel="stylesheet" href="font.css">
</head>
<body>
<p class="center">My text with font-arial</p>
<p class="right">My text with font-calibre</p>
<p class="left">My text with font-cooper</p>
</body>
</html>
```

font.css

```
p.center
{
color:pink;
text-align:center;
letter-spacing:3;
font-family:arial;
font-variant:small;
font-weight:bold;
```

```

}
p.right
{
color:orange;
text-align:right;
letter-spacing:10;
font-family:calibre;
font-variant:uppercase;
font-weight:bold;
}
p.left
{
color:green;
text-align:left;
letter-spacing:15;
font-family:cooper;
font-variant:uppercase;
font-weight:bold;
}

```

2) Set a background image

```

background.html
<html>
<head>
<style type="text/css">
body
{
background-image:url(wt.jpg);

background-repeat:repeat;
}
</style>
</head>
<body>
<h1 align="center">WebTechnology</h1>
</body>
</html>

```

3. Control the repetition of the image with the background-repeat property.

```

background-image.html
<html>
<head>
<style type="text/css">
body
{
background-image:url(wt.jpg);
background-color:pink;
background-repeat:no-repeat;
background-position:center;

```

```

background-attachment:fixed;
}
h1
{
background-image:url(wt1.jpg);
background-position:right;
color:red;
background-repeat:no-repeat;
}
</style>
</head>
<body>
<h1 align="center">WEB TECHNOLOGY<br></h1>
</body>
</html>

```

4) Define styles for links

Link.html

```

<html>
<head>
<style type="text/css">
a:link
{
color:red;
text-decoration:underline;
font-family:arial;
font-size:spot;
}
a:visited
{
color:yellow;
text-decoration:line-through;
font-family:arial;
font-size:spot;
}
a:hover
{
color:blue;
text-decoration:underline;
font-family:arial;
font-size:spot;
}
a:active
{
color:green;
text-decoration:underline;
font-family:arial;
font-size:spot;
}

```

```

</style>
</head>
<center>
<body bgcolor="pink">
<a href="wt.jpg" tagrget="_blank";>Link</a><br><br>
<a href="wt1.jpg" tagrget="_blank";>Link1</a><br><br>
</body>
</center>
</html>

```

5) Work with layers

layers.html

```

<html>
<body bgcolor="white" align="left">
<h1 style="background-color:blue; position:relative; top:0; left:0; z-index:2; height:100;"> This is
heading-1.</h1>
<h2 style="background-color:green; position:relative; top:30; left:50; z-index:3; height:100;"> This
is heading-2.</h2>
<h3 style="background-color:red; position:relative; top:55 left:100; z-index:1; height:100;"> This is
heading-3.</h3>
</body>
</html>

```

6) Add a customized cursor

cursor.html

```

<html>
<body>
<p style="cursor:not-allowed;">NOT-ALLOWED CURSOR</p>
<p style="cursor:progress;">PROGRESS CURSOR</p>
<p style="cursor:wait;">WAIT CURSOR</p>
<p style="cursor:zoom-in;">ZOOM-IN CURSOR</p>
<p style="cursor:zoom-out;">ZOOM-OUT CURSOR</p>
<p style="cursor:no-drop;">NO-DROP CURSOR</p>
<p style="cursor:move;">MOVE CURSOR</p>
<p style="cursor:default;">DEFAULT CURSOR</p>
<p style="cursor:scroll;">SCROLL CURSOR</p>
<p style="cursor:crosshair;">CROSSHAIR CURSOR</p>
<p style="cursor:help;">HELP CURSOR</p>
</body>
</html>

```

Execution procedure

Step1: type all html programs in notepad or any text editor and save .html format.

Step2: open web browser and open .html files.

Output

1) Use different font, styles:



2) Set a background image



3. Control the repetition of the image with the background-repeat property.



Control the repetition of the image with the background-repeat property

4) Define styles for links



Define styles for links

5) Work with layers



6) Add a customized cursor

NOT ALLOWED CURSOR
PROGRESS CURSOR
WAIT CURSOR
ZOOM-IN CURSOR
ZOOM-OUT CURSOR
NO DRAG CURSOR
MOVE CURSOR
DEFAULT CURSOR
SCROLL CURSOR
CROSSHAIR CURSOR
HELP CURSOR

Add a customized cursor

registration.html

```

<html>
<head>
<script type="text/javascript">
function checkfield(){
if(document.regfrm.fname.value.length==0){
alert("fill firstname field")
document.regfrm.fname.focus();
return;
}
if(document.regfrm.lname.value.length==0){
alert("fill lastname field")
document.regfrm.lname.focus();
return;
}
if(document.regfrm.eid.value.length==0){
alert("fill email-id")
document.regfrm.eid.focus();
return;
}
if(document.regfrm.zip.value.length==0){
alert("fill zip field")
document.regfrm.zip.focus();
return;
}
if(document.regfrm.phonenumber.value.length==0){
alert("fill phone number field")
document.regfrm.phonenumber.focus();
return;
}
else{
alert("Succesfull");
}
}
</script>
</head>
<body>
<center>
<form name="regfrm" method="post">
<p align="center">
<b>
<i><font face="monotype corsiva" color="black" size="7">
<u>Personal Registration Form</u>

```

```

</font>
</i>
</b>
</p>
<p align="center">
<font face="arial" color="bluishgreen" >
<b>You must complete fields marked with </b></font>
<font color="bluishgreen" face="times newroman">
*(star)</font></p>
<table border="8" cellpadding="0" cellspacing="0" width="73%" height="10">
<tr><td width="43%" height="22" >Firstname</font></td>
<td width="57%" height="22">
<input type="text" name="fname" size="20">
<font color="red"> * </font></td>

</tr>

<tr>

<td width="43%" height="22" >

<font face="timesnew roman" size="4">
Lastname
</font>
</td>

<td width="57%" height="22">
<input type="text" name="lname" size="20">
<font color="red"> * </font>

</td>

</tr>

<tr><td width="43%" height="22" >

<font face="timesnewroman" size="4">
Jobtitle
</font>
</td>

```

```

<td width="57%" height="22">
<font face="timesnewroman" size="20" >
<select size="1" name="jobsinfo">
<option value>choose..</option>
<option value="Engineer">Engineer</option>
<option value="Lecturer">Lecturer</option>
</select></font>
</td>
</tr>
<tr>
<td width="43%" height="22" >
<font face="timesnewroman" size="4">
Industry
</font>
</td>
<td width="57%" height="22">
<font face="timesnewroman"size="20" >
<select size="1" name="industry">
<option value>Choose..</option>
<option value="Academic">Academic</option>
<option value="Corporate">Corporate</option>
<option value="Financial">Financial</option>
<option value="Government">Government</option>
<option value="Military">Military</option>
<option value="Research">Research</option>
<option value="Technology">Technology</option>
<option value="Transportation">Transportation</option>
</select></font>
</td>

```

```

</tr>
<tr>
<td width="43%" height="22" >
<font face="timesnewroman" size="4">
Email-id
</font>
</td>
<td width="57%" height="22">
<input type="text" name="eid" size="20">
<font color="red"> * </font>
</td>
</tr>
<tr>
<td width="43%" height="22" >
<font face="timesnewroman" size="4">
Country
</font>
</td>
<td width="57%" height="38">
<font face="timesnewroman" size="20">
<select size="1" name="country">
<option value>Choose..</option>
<option value="IND">INDIA</option>
<option value="US">AMERICA</option>
<option value="JPN">JAPAN</option>
<option value="AUS">AUSTRALIA</option>
<option value="FRN">FRANCE</option>
</select></font>
</td>

```

```

</tr>
<tr>
<td width="43%" height="22" >
<font face="timesnewroman" size="4">
Organization
</font>
</td>
<td width="57%" height="22">
<input type="text" name="org" size="20">
</td>
</tr>
<tr>
<td ><font face="timesnewroman" size="4">
Address
</font>
</td>
<td>
<textarea rows="2" cols="16"></textarea>
</td>
</tr>
<tr>
<td width="43%" height="22" >
<font face="timesnewroman" size="4">
Town/city
</font>
</td>
<td width="57%" height="22">
<input type="text" name="city" size="20">
</td>

```

```

</tr>
<tr>
<td width="43%" height="22" >
<font face="timesnewroman" size="4">
State
</font>
</td>
<td width="57%" height="38">
<input type="text" name="state" size="20">
</td>
</tr>
<tr>
<td width="43%" height="38" >
<font face="timesnewroman" size="4">
zip/Postalcode
</font>
</td>
<td width="57%" height="38">
<input type="text" name="zip" size="20">
<font color="red">
*
</font>
</td>
</tr>
<tr>
<td width="43%" height="38" >
<font face="timesnewroman" size="4">
Phonenumber
</font>

```

```

</td>
<td width="57%" height="38">
<input type="text" name="phonenumber" size="20">
<font color="red">
*
</font>
</td>
</tr>
</table>
<p align="center"><font face="timesnewroman" color="bluishgreen">
<input type="radio" name="submit">
I agree terms and Conditions<br>
<input type="button" value="submit" name="b1" onclick="checkfield(">
<input type="reset" value="cancel" name="b2">
</p>
</form>
</center>
</body>
</html>

```

Execution Procedure

Step1: type the html program in notepad or any text editor and save .html format.

Step2: open web browser and open .html.

Output

LOGIN FORM VALIDATION

User Name: hellworld
Password: *****
Login RESET

Login Successful
Close

Login Page

Personal Registration Form

You must complete fields marked with *(star)

Firstname	CSE18Batch *
Lastname	GEC *
Jobtitle	Lecturer
Industry	Automic
Email-id	gocice18@gmail.com *
Country	INDIA
Organization	Gudlavallu Engineering C
Address	
Town/city	
State	
zip/Postalcode	521366 *
Phonenumber	9234567890 *

Successful
Close

I agree terms and Conditions
submit cancel

Registration Page

Personal Registration Form

You must complete fields marked with *(star)

Firstname	CSE19Rajath	*
Lastname	GEC	*
Jobtitle	Lecturer	
Industry	Academic	
Email-id		*
Country	INDIA	
Organization	Gudlavallu Engineering C	
Address	fill email-id	
Town/city	fill email-id	
State	Close	
zip/Postalcode	521318	*
Phonenumber	1234567890	*

I agree terms and Conditions

Registration Page – Email Validation

Personal Registration Form

You must complete fields marked with *(star)

Firstname	CSE19Rajath	*
Lastname	GEC	*
Jobtitle	Lecturer	
Industry	Academic	
Email-id	geccea19@gmail.com	*
Country	INDIA	
Organization	Gudlavallu Engineering C	
Address	fill phone number field	
Town/city	fill phone number field	
State	Close	
zip/Postalcode	521318	*
Phonenumber		*

I agree terms and Conditions

Registration Page – Phone Number Validation

Personal Registration Form

You must complete fields marked with *(star)

Firstname	CSE19Bach	*
Lastname	GEC	*
Jobtitle	Lecturer	
Industry	Academic	
Email-id	gecode19@gmail.com	*
Country	INDIA	
Organization	Gulbarga Engineering C	
Address	fill zip field	
Town/city		
State		Close
zip/Postalcode		*
Phonenumber	1234567890	*

I agree terms and Conditions

Registration Page – Postal Code Validation

EXP NO: 5

AIM: Write an XML file which will display the Book information which includes the following: Title of the book, Author Name, ISBN number, Publisher name, Edition and Price. Validate the above document using DTD and XML Schema.

Description

Extensible Markup Language (XML) is a meta-markup language that provides a format for describing structured data. This facilitates more precise declarations of content and more meaningful search results across multiple platforms. In addition, XML is enabling a new generation of Web-based data viewing and manipulation applications.

In the HTML you use tags to tell the browser to display data as bold or italic; in XML you use tags only to describe data, such as city name, temperature, and barometric pressure. In XML, you use style sheets such as Extensible Stylesheet Language (XSL) and Cascading Style Sheets (CSS) to present the data in a browser. XML separates the data from the presentation and the process, enabling you to display and process the data as you wish by applying different style sheets and applications.

Source Code

a) Demonstrating DTD file

bookstore.dtd

```
<!ELEMENT bookstore (book+)>
<!ELEMENT book (title,author+,ISBN,publisher,edition,prices)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT author (#PCDATA)>
<!ELEMENT ISBN (#PCDATA)>
<!ELEMENT publisher (#PCDATA)>
<!ELEMENT edition (#PCDATA)>
<!ELEMENT prices (#PCDATA)>
```

books.xml

```
<?xml version="1.0"?>
<!DOCTYPE bookstore SYSTEM "bookstore.dtd">
<bookstore>

    <book>
        <title>Web Programming, Building Internet Applications</title>
        <author>Chris Bates </author>
        <ISBN>123-456-789</ISBN>
        <publisher>Wiley</publisher>
        <edition>2nd</edition>
        <prices>329</prices>
    </book>

    <book>
        <title>Web Technologies, Black book</title>
        <author>Kogent</author>
```

```

    <ISBN>123-456-789</ISBN>
    <publisher>Dreamtech</publisher>
    <edition>2nd</edition>
    <prices>229</prices>
  </book>

  <book>
    <title>WEB TECHNOLOGIES</title>
    <author>Uttam.K.Roy</author>
    <ISBN>123-456-789</ISBN>
    <publisher>Oxford</publisher>
    <edition>2nd</edition>
    <prices>429</prices>
  </book>

```

```
</bookstore>
```

b) Demonstrating XSD

book-schema.xsd

```

<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="bookstore">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="book">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="title" type="xs:string"/>
              <xs:element name="author" type="xs:string"/>
              <xs:element name="ISBN" type="xs:int"/>
              <xs:element name="publisher" type="xs:string"/>
              <xs:element name="edition" type="xs:int"/>
              <xs:element name="prices" type="xs:int"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Book-schema.xsl

```

<?xml version="1.0"?>
<bookstore xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="book-schema.xsd">

    <book>
        <title>Web Programming, Building Internet Applications</title>
        <author>Chris Bates </author>
        <ISBN>123456789</ISBN>
        <publisher>Wiley</publisher>
        <edition>2</edition>
        <prices>329</prices>
    </book>

    <book>
        <title>Web Technologies, Black book</title>
        <author>Kogent</author>
        <ISBN>123-456-789</ISBN>
        <publisher>Dreamtech</publisher>
        <edition>2nd</edition>
        <prices>229</prices>
    </book>

    <book>
        <title>WEB TECHNOLOGIES</title>
        <author>Uttam.K.Roy</author>
        <ISBN>123-456-789</ISBN>
        <publisher>Oxford</publisher>
        <edition>2nd</edition>
        <prices>429</prices>
    </book>
</bookstore>

```

Execution Procedure

Step1: Type all html programs in notepad or any text editor and save the files.

Step2: open appropriate xml files in web browser for getting output.

Output

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

<bookstore>
  <book>
    <title>Web Programming, Building Internet Applications</title>
    <author>Chris Bates </author>
    <ISBN>123-456-789</ISBN>
    <publisher>Wiley</publisher>
    <edition>2nd</edition>
    <prices>329</prices>
  </book>
  <book>
    <title>Web Technologies, Black book</title>
    <author>Robert </author>
    <ISBN>123-456-789</ISBN>
    <publisher>Dreamtech</publisher>
    <edition>2nd</edition>
    <prices>229</prices>
  </book>
  <book>
    <title>WEB TECHNOLOGIES</title>
    <author>Uttam.K.Roy </author>
    <ISBN>123-456-789</ISBN>
    <publisher>Oxford</publisher>
    <edition>2nd</edition>
    <prices>429</prices>
  </book>
</bookstore>

```

Demonstrating DTD

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

<bookstore xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:bookspace="http://book-space.xsd">
  <book>
    <title>Web Programming, Building Internet Applications</title>
    <author>Chris Bates </author>
    <ISBN>123456789</ISBN>
    <publisher>Wiley</publisher>
    <edition>2</edition>
    <prices>329</prices>
  </book>
  <book>
    <title>Web Technologies, Black book</title>
    <author>Robert </author>
    <ISBN>123-456-789</ISBN>
    <publisher>Dreamtech</publisher>
    <edition>2nd</edition>
    <prices>229</prices>
  </book>
  <book>
    <title>WEB TECHNOLOGIES</title>
    <author>Uttam.K.Roy </author>
    <ISBN>123-456-789</ISBN>
    <publisher>Oxford</publisher>
    <edition>2nd</edition>
    <prices>429</prices>
  </book>
</bookstore>

```

Demonstrating XSD

EXP NO: 6**AIM:**

- i. Write a PHP program to validate the fields of the login page.
- ii. Write a PHP program to validate the fields of the Registration page

DESCRIPTION:

This PHP login application uses MySQL database to store user information and all the input parameters are validated with javascript. This PHP login page using MySQL database connections contains PHP 7.0 methods only. Many methods like `mysql_real_escape_string()`, `mysql_query()`, `mysql_num_rows()`, `mysql_connect()`, `mysql_select_db()`, `mysql_close()` were deprecated in PHP 5 and removed from the latest version of PHP i.e. PHP 7.0.

ALGORITHM:**LoginForm.php**

- Step 1:-Open the html tag and the title as LoginForm.php.
- Step 2:-Declare the validate() function.
- Step 3:-Design login form with username, password, submit and reset values.
- Step 4:-if anyone not enter user name or password it shows alert message.
- Step 5:-close the html tag.

Login.php

- Step 1:-Open the html and body tag.
- Step 2:-Open the MYSQL database connection.
- Step 3:-if the `$_SERVER['REQUEST_METHOD'] == 'POST'` then
- Step 4:-enter user name and password
- Step 5:-check the user name and password is valid or not
- Step 6:-if valid then display the login page
- Step 7:-otherwise it shows incorrect username and password.
- Step 8:-close body and html tags.

PROGRAM:**login.html**

```
<html>

    <body bgcolor=azure>
        <form name="LoginForm" action="http://localhost:2424/19batch/login-
validate.php" method="post">
            <fieldset>
                <legend align=center><h2>Login Here</h2></legend>
                <table cellpadding=5 align=center>
                    <tr>
                        <td>User Name</td>
                        <td> : </td>
                        <td><input type="text" name="t1" class="textbox"
placeholder="Only Characters" /></td>
                    </tr>
                    <tr>
                        <td> Password </td>
                        <td> : </td>
```

```

                <td><input type="Password" name="t2" class="textbox"
placeholder="Max 15 characters"/></td>
            </tr>
            <tr>
                <td ><input type="submit" class="button button1"
value="LOGIN" /></td>
                <td></td>
            <tr>
                <td ><input type="reset" class="button button1"
value="RESET" /></td>
                <td></td>
            </tr>
        </table>
    </form>
</body>
</html>

```

Login-validation.php

```

<?php
    $user=$_POST['t1'];
    $pwd=$_POST['t2'];
    $uname_pat='/^[a-zA-Z0-9]{5,13}$/';
    $pwd_pat='/^[a-zA-Z0-9@$$ ]{5,13}$/';
    $flag=true;
    if($user=="") {
        $flag=false;
        echo "PLEASE Enter the user name ";
    }
    else if($pwd=="") {
        $flag=false;
        echo "PLEASE Enter the password ";
    }
    else if(preg_match($uname_pat,$user)==false) {
        $flag=false;
        echo "Please Enter UserName Format";
    }
    else if(preg_match($pwd_pat,$pwd)==false) {
        $flag=false;
        echo "Please Enter Password Format";
    }
    else if($flag==true) {
        echo "All Details are valid";
    }
}
?>

```

registration.php

```

<html>

    <body bgcolor=azure>

```



```

    <form name="RegistrationPage"
action="http://localhost:2424/19batch/registration-validation.php" method="post">
    <fieldset>
    <legend align=center><h2>REGISTER HERE</h2></legend>
    <table cellpadding=5 align=center >
        <tr>
            <td>First Name</td>
            <td> : </td>
            <td><input type="text" name="t1" class ="textbox"
placeholder="Only Characters" /></td>
        </tr>
        <tr>
            <td>LastName</td>
            <td> : </td>
            <td><input type="Password" name="t2" class ="textbox"
placeholder="Max 15 characters"/></td>
        </tr>
        <tr>
            <td> Password</td>
            <td> : </td>
            <td><input type="Password" name="t3" class ="textbox"
placeholder="Max 15 characters"/></td>
        </tr>
        <tr>
            <td>Contact Number</td>
            <td> : </td>
            <td><input type="text" name="t4" class ="textbox"
placeholder="Without country code" max=10/></td>
        </tr>
        <tr>
            <td>Email Address</td>
            <td> : </td>
            <td><input type="text" name="t5" class ="textbox"
placeholder="Mail ID" max=35/></td>
        </tr>
        <tr>
            <td>Address</td>
            <td> : </td>
            <td>
                <input type="text" name="t6" class ="textbox" rows=5 cols=50
                </input>
            </td>
        </tr>
        <tr>
            <td>Gender</td>
            <td> : </td>
            <td><input type="radio" name="gender" value="male"
checked> Male<br>
            <input type="radio" name="gender" value="female"
checked> Female</td>
        </tr>
    </table>
    </fieldset>
    </form>

```

```

Female<br>
Other
value="REGISTER" /></td>
value="RESET" /></td>
<input type="radio" name="gender" value="female">
<input type="radio" name="gender" value="other">
</td>
</tr>
<tr>
<td ><input type="submit" class="button button1"
value="REGISTER" /></td>
<td></td>
<td ><input type="reset" class="button button1"
value="RESET" /></td>
</tr>
</table>
</form>
</body>
</html>

```

registration-validation. php

```

<?php
$fn=$_POST['t1'];
$ln=$_POST['t2'];
$pwd=$_POST['t3'];
$mob=$_POST['t4'];
$mail=$_POST['t5'];
$fn_pat='/^[a-zA-Z][a-zA-Z0-9]{5,13}$/';
$ln_pat='/^[a-zA-Z][a-zA-Z0-9]{5,13}$/';
$pwd_pat='/^[a-zA-Z0-9@& ]{5,13}$/';
$mob_pat='/^\d{10}$/';
$mail_pat='/^[A-Za-z0-9]+@[A-Za-z]+\.[A-Za-z]{2,4}$/';
$flag=true;
if($fn=="") {
    $flag=false;
    echo "Enter the First name PLEASE";
}
else if($ln=="") {
    $flag=false;
    echo "Enter the LastPLEASE";
}
else if($pwd=="") {
    $flag=false;
    echo "Enter the password PLEASE";
}
else if($mob=="") {
    $flag=false;
    echo "Enter the Mobile NumberPLEASE";
}
else if($mail=="") {

```

```

        $flag=false;
        echo "Enter the E-Mail PLEASE";
    }
    else if(preg_match($fn_pat,$fn)==false) {
        $flag=false;
        echo "Please Enter First Name in Given Format";
    }
    else if(preg_match($ln_pat,$ln)==false) {
        $flag=false;
        echo "Please Enter Last Name in Given Format";
    }
    else if(preg_match($pwd_pat,$pwd)==false) {
        $flag=false;
        echo "Please Enter Password in Given Format";
    }


    else if(preg_match($mob_pat,$mob)==false) {
        $flag=false;
        echo "Please Enter Mobile in Given Format";
    }
    else if(preg_match($mail_pat,$mail)==false) {
        $flag=false;
        echo "Please Enter Mail in Given Format";
    }
    else if($flag==true) {
        echo "All details are valid";
    }
}

```

?>

Output:





The image shows a web browser window with a registration form. The browser's address bar shows the URL `localhost:3030/15batch/register.html`. The form is titled "REGISTER HERE" and contains the following fields and options:

- First Name:
- Last Name:
- Password:
- Contact Number:
- Email Address:
- Address:
- Gender: Male, Female, Other

At the bottom of the form are two buttons: "REGISTER" and "RESET".

EXP NO: 7

AIM: Write a JSP to connect to the database and extract data from the tables and display them to the user.

DESCRIPTION**Source Code:****Retrieve.Html**

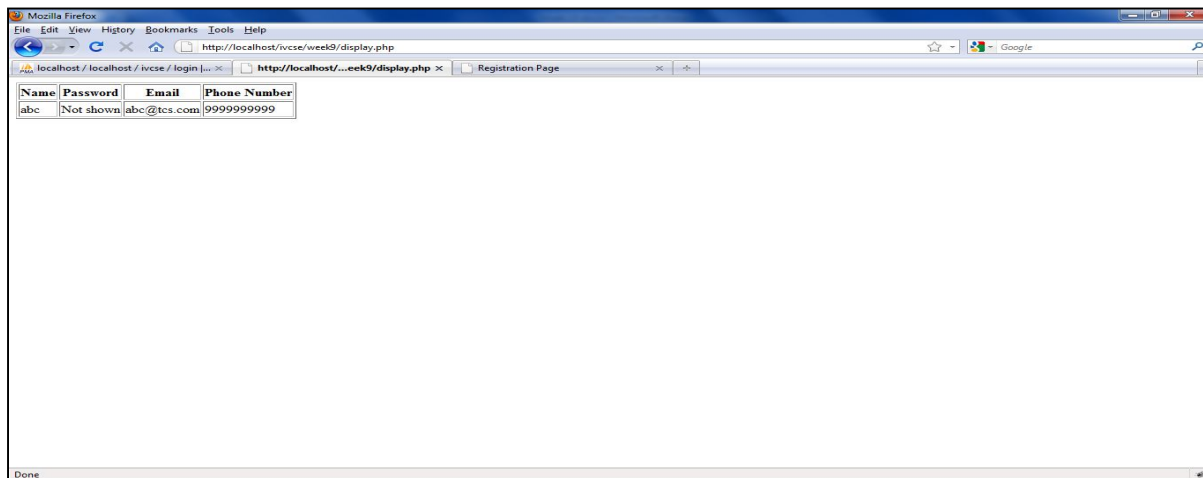
```
<html>
<head><title>loginpage</title></head>
<body>
<center><br><br><br><h1><font size="22" color="pink"><i>Login Page</font></h1></i>
<form action="retrieve.jsp" name="log" id="li" onSubmit="return validate( )" action="#"
method="post">
<table><tr><td>Login Id:</td><td><input type="text" name="login" size="10"><span
id="s1"></span></td></tr>
<tr><td>Password:</td><td><input type="password" name="pwd" size="10"><span
id="s2"></span></td></tr>
<tr><td><input type="submit" value="Login"></td><td><input type="reset"
value="reset"></td></tr></table>
</form>
</body>
</html>
```

retrieve.jsp

```
<% @page import="java.sql.*"%>
<html>
<body>
<%
String uname=request.getParameter("login");
String pwd=request.getParameter("pwd");
try{
Class.forName("com.mysql.jdbc.Driver");
Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/srimanth","root","");
Statement st=con.createStatement();
PreparedStatement p=con.prepareStatement("select * from student");
p.setString(1,uname);
p.setString(2,pwd);

if(uname==E_mail&&pwd==Password) {
response.sendRedirect("welcome.jsp");
}
}
```

OUTPUT:



EXP NO: 8

AIM: Design a JSP to insert the details of the users who register through the registration page and store the details in to the database.

PROGRAM:

Registration.html

```
<html>
<head><title>register</title></head>
<body>
<center><br><br><br><h1><font size="22" color="pink"><i>Registration</font></h1></i>
<form action="registration.jsp" method="post" name="reg" id="rn" onSubmit="return validate( )">
<table><tr><td>Name:</td><td><input type="text" name="na" size="10"></td></tr>
<tr><td>Gender:</td><td><input type="Radio" name="gender" >Male<input type="Radio"
name="gender" >Female</td></tr>
<tr><td>Date of birth:</td><td><input type="date" name="dob" size="10"></td></tr>
<tr><td>Rollno:</td><td><input type="text" name="roll" size="10"></td></tr>
<tr><td>Phone no:</td><td><input type="text" name="pno" size="10"></td></tr>
<tr><td>Email id:</td><td><input type="text" name="mail" size="30"></td></tr>
<tr><td>Password:</td><td><input type="password" name="pwd" size="10"></td></tr>
<tr><td>Re-enter Password:</td><td><input type="password" name="rpwd" size="10"></td></tr>
<tr><td>Branch:</td><td><select name="br"><option value="default">Select
branch</option><option value="IT">Information Technology</option><option
value="CSE">Computer Science Engineering</option><option value="ECE">Electronics and
Communication Engineering</option><option value="EEE">Electrical and Electronics
Engineering</option><option value="CIVIL">Civil Engineering</option><option
value="MECH">Mechanical Engineering</option></td></tr>
<tr><td>College:</td><td><input type="text" name="col" size="10"></td></tr>
<tr><td>Address:</td><td><textarea name="add" rows="4" cols="30"></textarea></td></tr>
<tr><td><input type="submit" value="Register"></td><td><input type="reset"
value="reset"></td></tr></table>
</form>
</body>
</html>
```

registration.jsp

```
<% @page import="java.sql.*"%>
<html>
<body>
<%
String Name=request.getParameter("na");
String Gender=request.getParameter("gender");
String Dob=request.getParameter("dob");
String Rollno=request.getParameter("roll");
String Phoneno=request.getParameter("pno");
String E_mail=request.getParameter("mail");
String Password=request.getParameter("pwd");
String Branch=request.getParameter("br");
String College=request.getParameter("col");
```

```

String Address=request.getParameter("add");
try{
Class.forName("com.mysql.jdbc.Driver");
Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/srimanth","root","");
Statement st=con.createStatement();
PreparedStatementps=con.prepareStatement("insert into
student(Name,Gender,Dob,Rollno,Phoneno,E_mail>Password,Branch,College,Address)values(?,?,?,
?,?,?,?,?,?)");
ps.setString(1,Name);
ps.setString(2,Gender);
ps.setString(3,Dob);
ps.setString(4,Rollno);
ps.setString(5,Phoneno);
ps.setString(6,E_mail);
ps.setString(7>Password);
ps.setString(8,Branch);
ps.setString(9,College);
ps.setString(10,Address);
int x=ps.executeUpdate();
%><a href="welcome.jsp">Register</a><%
}
catch(Exception e) {
out.print(e.getMessage());
}
%>
</body>
</html>

```

welcome.jsp

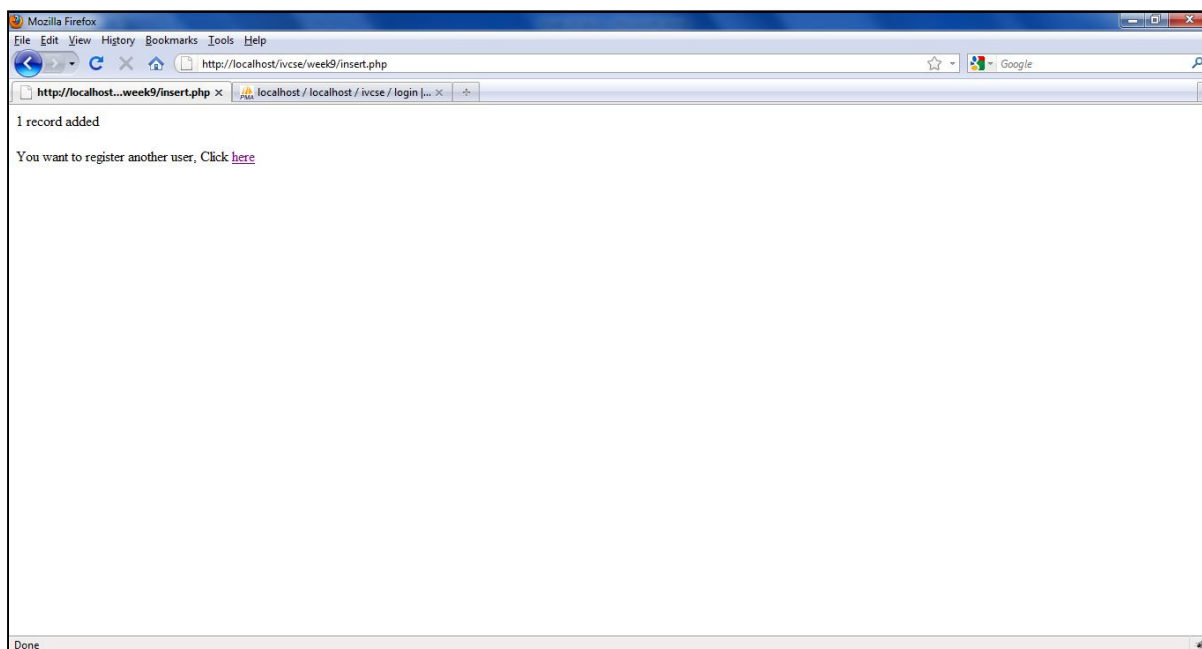
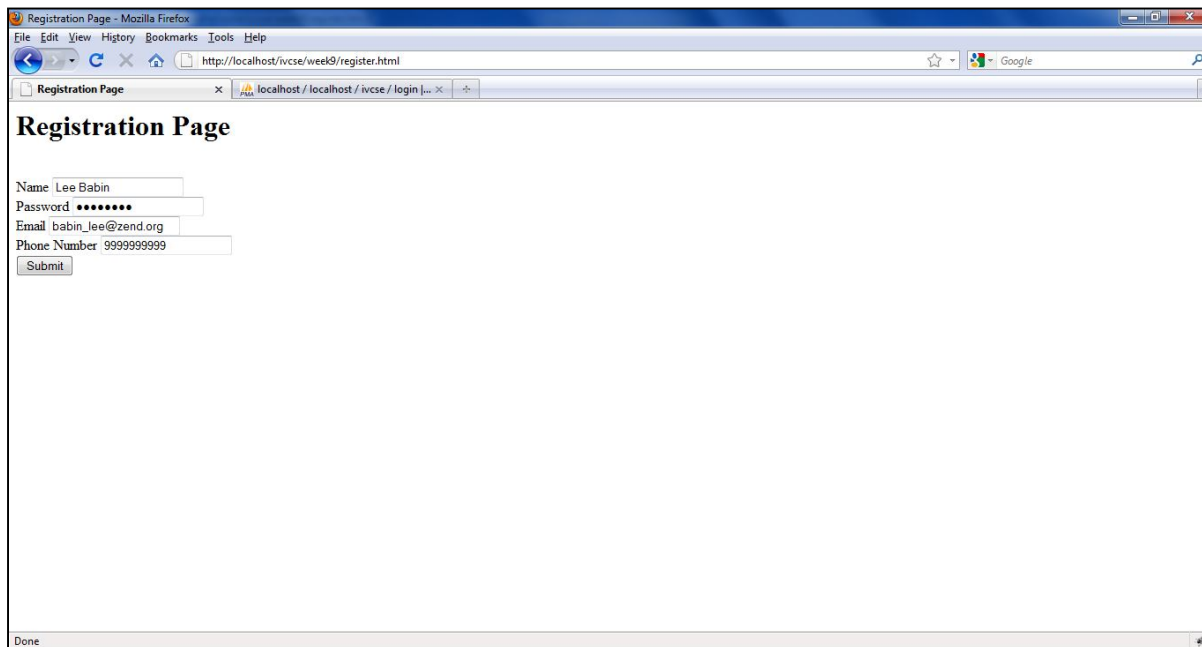
```

<html>
<body bgcolor="green"><center>
<font size="22" color="red">
<%
out.print("1 record added");
out.print("<br> You want to register another user");

%>
Click <a href="Registration.html">here</a>
</font>
</center>
</body>
</html>

```

OUTPUT:



EXP NO: 9

AIM:

Write a PHP program to connect to MySQL database which retrieves the data from the tables and display them to the user.

Description

Create a Connection to a MySQL Database

Before you can access data in a database, you must create a connection to the database.

In PHP, this is done with the `mysqli_connect()` function.

Syntax

```
mysqli_connect(servername,username,password,dbname);
```

Parameter	Description
Servername	Optional. Specifies the server to connect to. Default value is "localhost:3306"
Username	Optional. Specifies the username to log in with. Default value is the name of the user that owns the server process
Password	Optional. Specifies the password to log in with. Default is ""
Dbname	Database Name

Example

In the following example we store the connection in a variable (\$con) for later use in the script. The "die" part will be executed if the connection fails:

```
<?php
$con = mysqli_connect("localhost","root","","srikanth");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

// some code
?>
```

Closing a Connection

The connection will be closed automatically when the script ends. To close the connection before, use the `mysqli_close()` function:

```
<?php
$con = mysqli_connect("localhost","root","","srikanth");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}

// some code

mysqli_close($con);
?>
```

Source code

Step by step procedure for creating table in php my admin

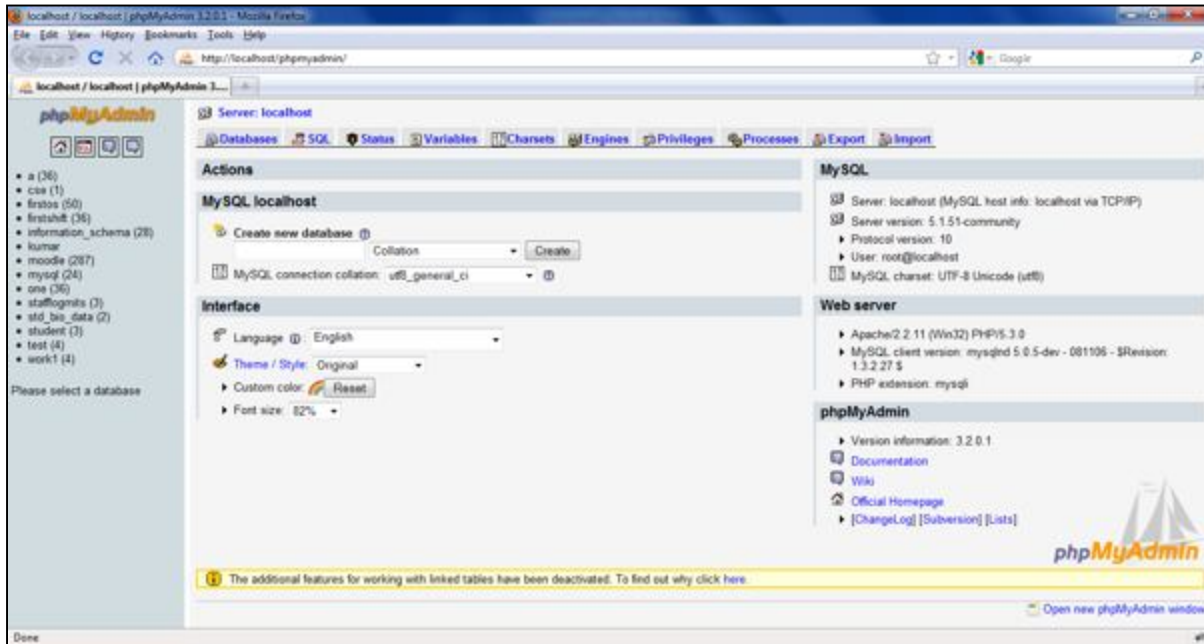
MySQL is automatically installed along with WAMP server installation.
User environment for MySQL in WAMP is “phpMyAdmin”.

Step1: Start WAMP server

Step2: Click WAMP server icon on taskbar and select “phpMyAdmin” option.

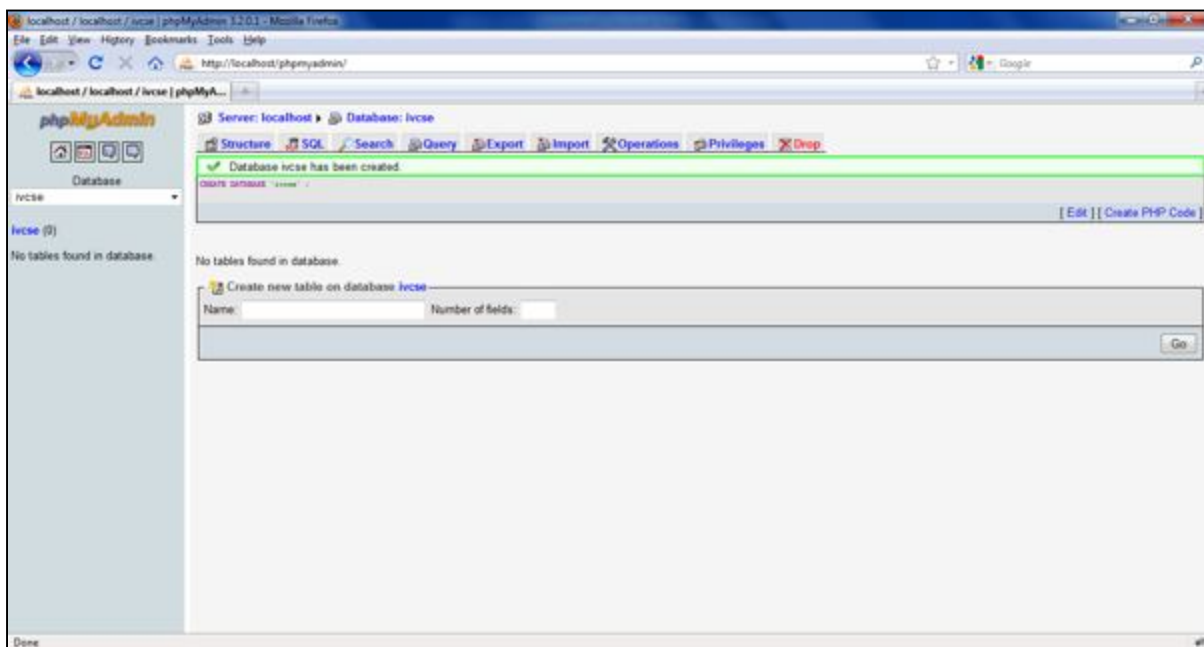


It automatically opens “phpMyAdmin” screen.

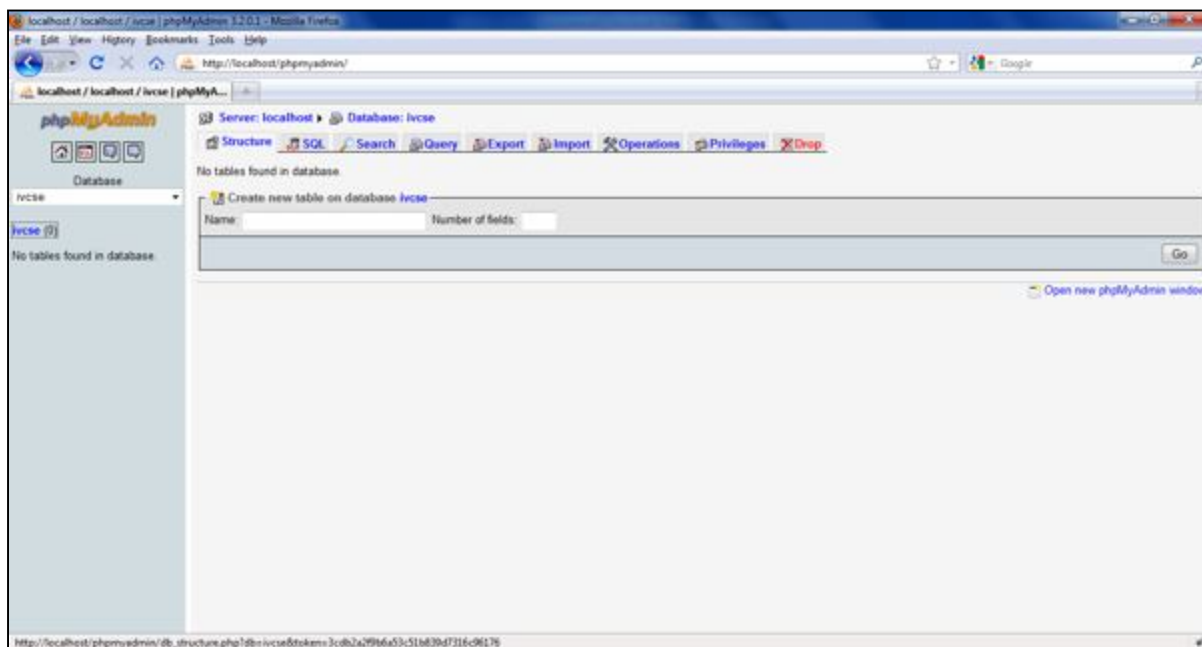


Step3: create New Database.

Provide database name on “create new database” field and click “create” button. The following screen displays in your computer



Step 4: Click database “ivcse” i.e appeared left panel of phpMyAdmin Screen. It displays following screen

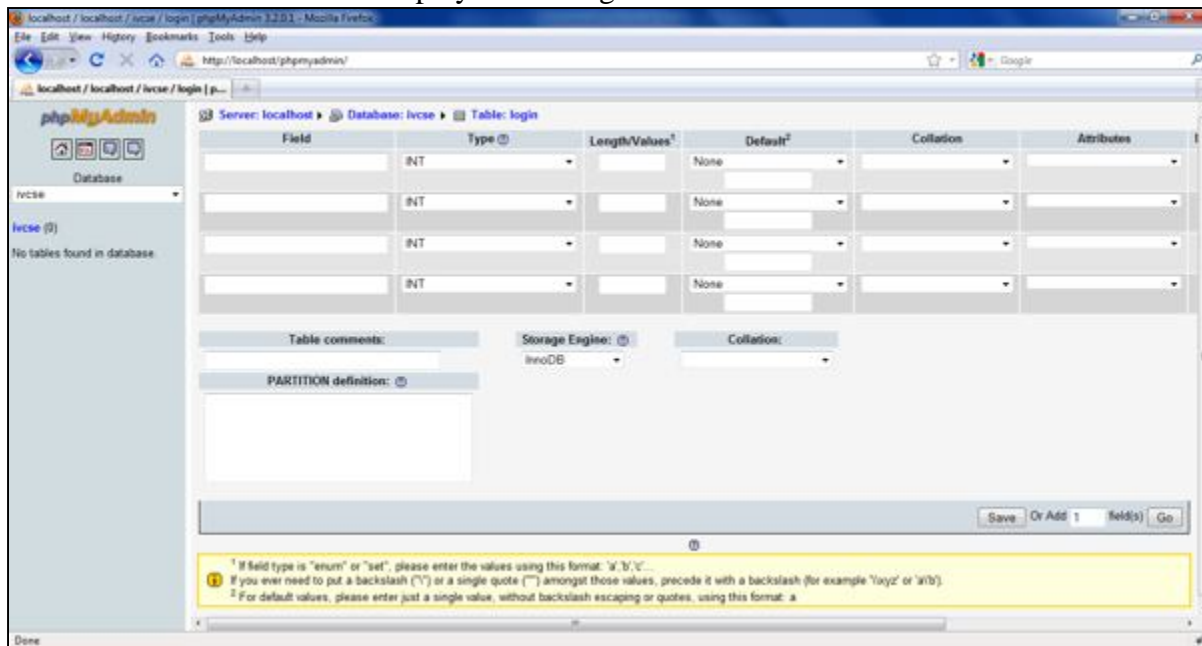


Step5: Creating a table.

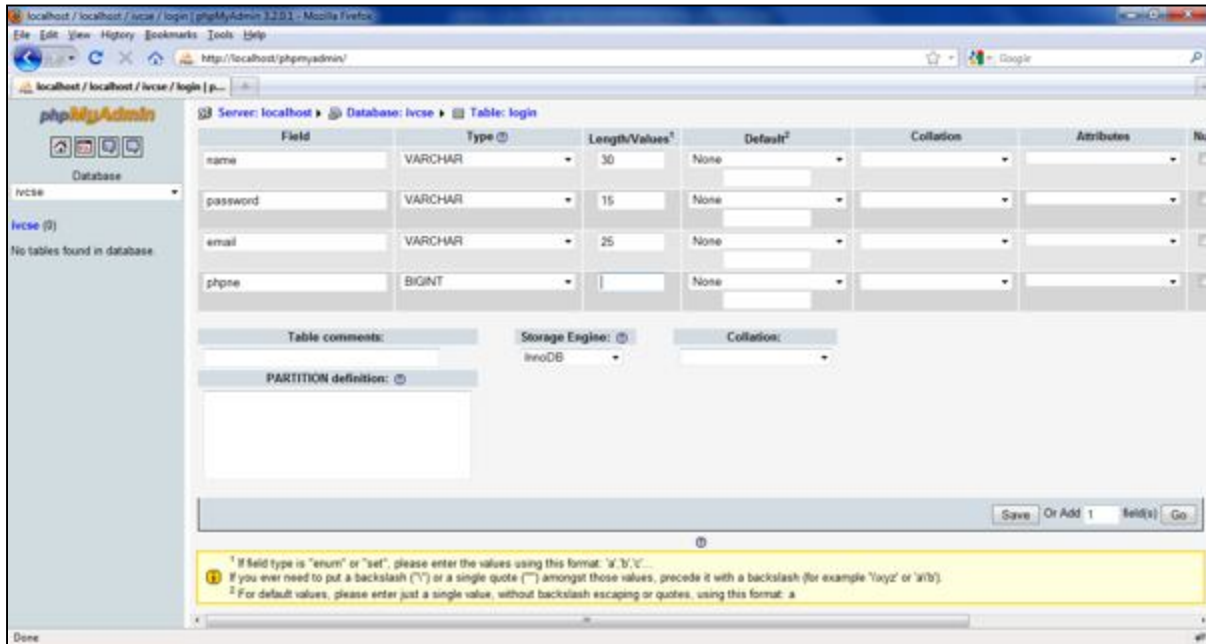
The Aim indicates that you can create which should contain at least the following fields: name, password, email-id, phone number. So here 4 fields are there.

Provide table name “login” and “number of fields:” 4

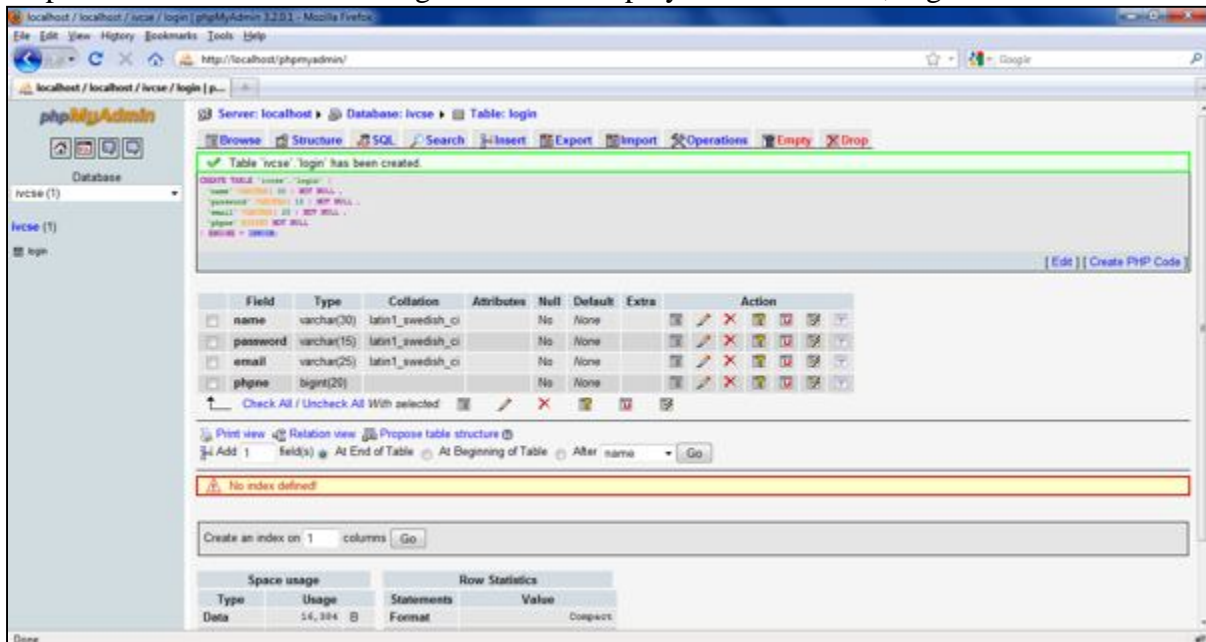
Click “Go” button. It displays following screen



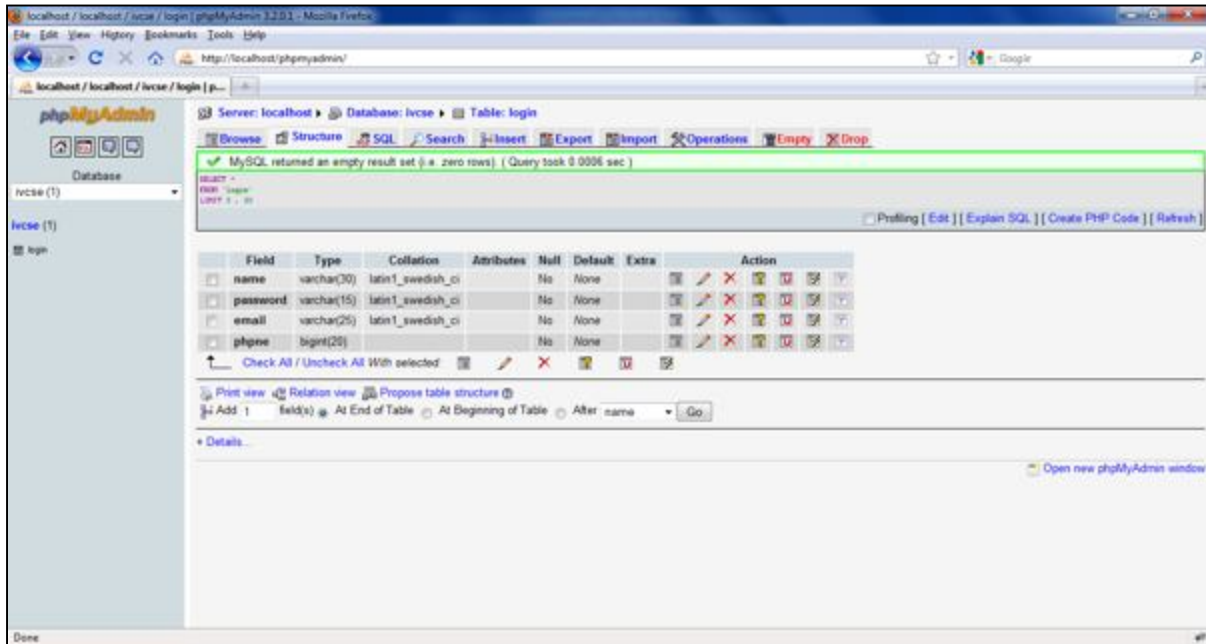
Step 6: Provide fields name and its datatype, size like the following screen



Step 7: click “Save”. It create login table and displays “Table ‘ivcse’,’login’ has been created.”



Step 8: click on table name on left panel of phpMyAdmin, you can see structure of table i.e field name datatype and other parameters



```

Display.php
<?php
$con = mysqli_connect("localhost","root","","srikanth");
if (!$con)
{
    die('Could not connect: ' . mysqli_error());
}

mysql_select_db("ivcse", $con);

$result = mysqli_query($con, "SELECT * FROM login");

echo "<table border='1'>
<tr>
<th>Name</th>
<th>Password</th>
<th>Email</th>
<th>Phone Number</th>
</tr>";

while($row = mysqli_fetch_array($result))
{
    echo "<tr>";
    echo "<td>" . $row['name'] . "</td>";
    echo "<td> Not shown</td>";
    echo "<td>" . $row['email'] . "</td>";
    echo "<td>" . $row['phno'] . "</td>";
    echo "</tr>";
}
echo "</table>";

```



```
mysqli_close($con);  
?>
```

Execution procedure

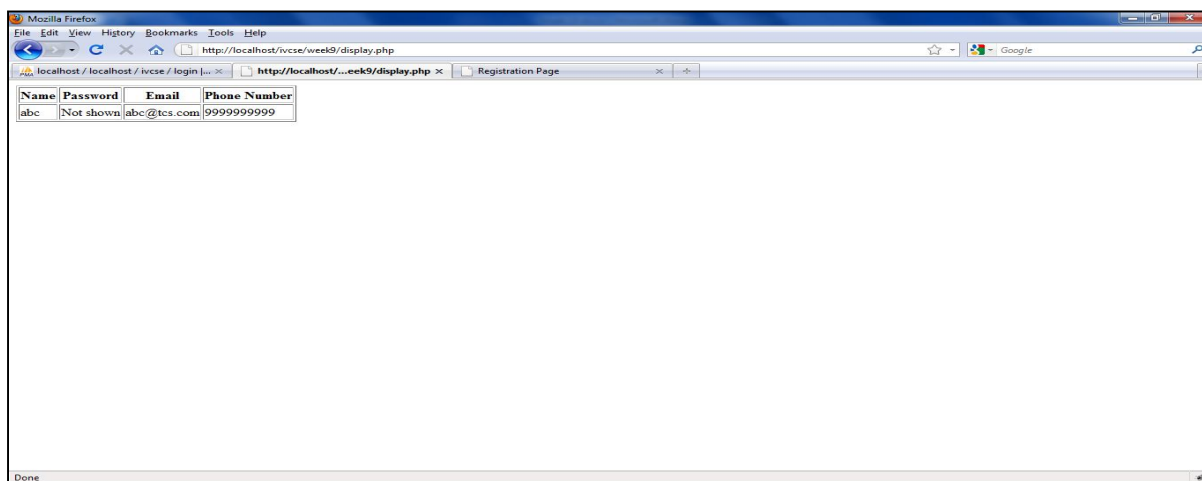
Step1: Type all programs and save with .php extension in c:\WAMP\WWW directory.

Step2: Run WAMP Server.

Step3: Through localhost execute your programs.

url's are: <http://localhost/display.php>&&<http://localhost/insert.php>

Output



Displaying data from Database

Registering New User

After providing all fields of data and Click on Submit button

After successfully added 1 user

EXP NO: 10**AIM:**

Write a PHP program to insert the details entered by the user in the Registration form into MySQL database.

DESCRIPTION:**PROGRAM-1:****Register.php**

```
<html>
<title>Registration Page</title>
</head>
<body>
<h1> Registration Page</h1>
<br />
<form action="insert.php" method="post">
Name <input type="text" name="fname" id="fname" value="name" /><br />
Password <input type="password" name="pwd" id="pwd" value="password" /><br />
Email <input type="text" name="email" id="email" value="email@example.com"/><br />
Phone Number <input type="text" name="ph" id="ph" value="9999999999" maxlength="10"/><br />
<input type="submit" value="Submit" />
</form>
</body>
</html>
```

Insert .php

```
<?php
$con = mysqli_connect("localhost","root","root","srikanth");
if (!$con)
{
    die('Could not connect: ' . mysqli_error());
}

mysql_select_db("ivcse", $con);

$sql="INSERT INTO login (name, password, email, phno)
VALUES
('$_POST[fname]','$_POST[pwd]','$_POST[email]','$_POST[ph])";

if (!mysqli_query($con,$sql))
{
    die('Error: ' . mysqli_error());
}
echo "1 record added";
echo "<br><br>";
echo "You want to register another user, Click <a href='register.html'>here</a>";
```

```
mysqli_close($con);  
?>
```

Output:

