MAHATMA GANDHI MISSION

In Pursuit of Excellence

Education | Service | Research

MGM's

Jawaharlal Nehru Engineering College Aurangabad

Affiliated to Dr.B.A.Technological University, Lonere Maharashtra

ISO 9001:2015, 140001:2015 Certified, AICTE Approved

Department of Computer Science & Engineering

LAB MANUAL

Programme (UG/PG)	: UG
Year	: Final Year
Semester	: VIII
Course Code	: BTCOL705
Course Title	: Full Stack Developer

Prepared By
M K Ugale
Assistant Professor
Department of Computer Science & Engineering

FOREWORD

It is my great pleasure to present this laboratory manual for Final year engineering students for the subject of Full Stack Development

As a student, many of you may be wondering with some of the questions in your mind regarding the subject and exactly what has been tried is to answer through this manual.

As you may be aware that MGM has already been awarded with ISO 9001:2015,140001:2015 certification and it is our endure to technically equip our students taking the advantage of the procedural aspects of ISO Certification.

Faculty members are also advised that covering these aspects in initial stage itself, will greatly relived them in future as much of the load will be taken care by the enthusiasm energies of the students once they are conceptually clear.

Dr. H. H. Shinde
Principal

LABORATORY MANUAL CONTENTS

This manual is intended for the Final year students of Computer Science & Engineering in the subject of Full Stack Development. This manual typically contains practical/Lab Sessions related to Full Stack Development covering various aspects related the subject to enhanced understanding.

Students are advised to thoroughly go through this manual rather than only topics mentioned in the syllabus as practical aspects are the key to understanding and conceptual visualization of theoretical aspects covered in the books.

Good Luck for your Enjoyable Laboratory Sessions

Mr.M K Ugale	**Dr. Vijaya Musande**
Subject Teacher	HOD

LIST OF EXPERIMENTS

Course Code: BTCOL705
Course Title: Full Stack Development.

Sr. No	Name of the Experiment	Page No
	Basics-Introduction	
1	Explore Basic HTML Tags and Elements.	
2	Familiarize with **JS, and CSS, Animation** using sample webpage	
	HTML,CSS,JAVASCRIPT,	
3.	Design a Webpage using advance HTML Form tags input–date, time, number, email, HTML5 Header And Footer, spell check and editable areas.	
4.	Design a Webpage Demonstrating Drag and Drop Functionality. Implement program demonstrating Local Storage and session storage.	
5.	Design a Webpage using Basic CSS Tags. Demonstrate Inline, Internal and External Style sheets using advanced CSS.	
6.	Design signup form to validate username, password, and phone numbers etc using Java script. Write a program to demonstrate Event Handling using JavaScript.(Minimum 3Events)	
7.	Design a Form using HTML and CSS and accept the data from it and insert it into Database using PHP.	
8.	Create a form with a text box asking to enter your favorite city with a submit button when the user enters the city and clicks the submit button another PHP page should be opened displaying “Welcome to the city”.	
9.	Change a Content of webpage using AJAX. Perform Different Operations using JQUERY Selectors.	
10.	Mini Project	

DOs and DON'Ts in Laboratory:

1. Make entry in the Log Book as soon as you enter the Laboratory.

2. All the students should sit according to their roll numbers starting from their left to right.

3. All the students are supposed to enter the terminal number in the log book.

4. Do not change the terminal on which you are working.

5. All the students are expected to get at least the algorithm of the program/concept to be implemented.

6. Strictly observe the instructions given by the teacher/Lab Instructor.

7. Do not disturb machine Hardware / Software Setup.

Instruction for Laboratory Teachers:

1. Submission related to whatever lab work has been completed should be done during the next lab session along with signing the index.

2. The promptness of submission should be encouraged by way of marking and evaluation patterns that will benefit the sincere students.

3. Continuous assessment in the prescribed format must be followed.

MGM's

In Pursuit of Excellence

Education | Service | Research

Jawaharlal Nehru Engineering College, Aurangabad

Department of Computer Science and Engineering

Vision of CSE Department

To develop computer engineers with necessary analytical ability and human values who can creatively design, implement a wide spectrum of computer systems for welfare of the society.

Mission of the CSE Department:

~~1.~~ Preparing graduates to work on multidisciplinary platforms associated with their professional position both independently and in a team environment.

~~2.~~ Preparing graduates for higher education and research in computer science and engineering enabling them to develop systems for society development.

Programme Educational Objectives

Graduates will be able to

I. To analyze, design and provide optimal solution for Computer Science & Engineering and multidisciplinary problems.

II. To pursue higher studies and research by applying knowledge of mathematics and fundamentals of computer science.

III. To exhibit professionalism, communication skills and adapt to current trends by engaging in lifelong learning.

Programme Outcomes (POs):

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 independent and life-long learning in the broadest context of technological change.

LABORATORY OUTCOMES

The practical/exercises in this section are psychomotor domain Learning Outcomes (i.e. subcomponents of the COs), to be developed and assessed to lead to the attainment of the competency.

LO-1: Study and Implement WebPages using Basic and Advanced HTML.

LO-2: Differentiate between functionalities of Basic CSS and Advanced CSS.

LO-3: Implement basic JavaScript.

LO-4: Design Webpages using Ajax, jQuery, PHP, PHP Advanced.

1. Lab Exercise

Exercise No 1: (2 Hours) – 1 Practical

AIM: - Explore Basic HTML Tags and Elements.

OBJECTIVES:

1. Define HTML and common terminology related to HTML.
2. Recognize correct HTML syntax.
3. Students should able to install to write a brief, error-free HTML code.

PROBLEM STATEMENT:

1. Create a Departmental Info webpage using Basic HTML Tags and Elements.

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.

TAGS TO BE USED:

- Formatting Tags
- Forms and Input tags
- Frames
- Images
- Audio/Video
- Links
- Lists
- Tables
- Styles and Semantics

For information use reference-

https://www.w3schools.com/tags/

CONCLUSIONS:

In this practical we have studied Basic HTML Tags and Elements.

2. Lab Exercise

Exercise No 2: (2 Hours) – 1 Practical

Aim: - Familiarize with JS, and CSS, Animation using sample webpage

Objectives:

- Deep Dive Into Modern Web Development
- The Full-Stack Web Development

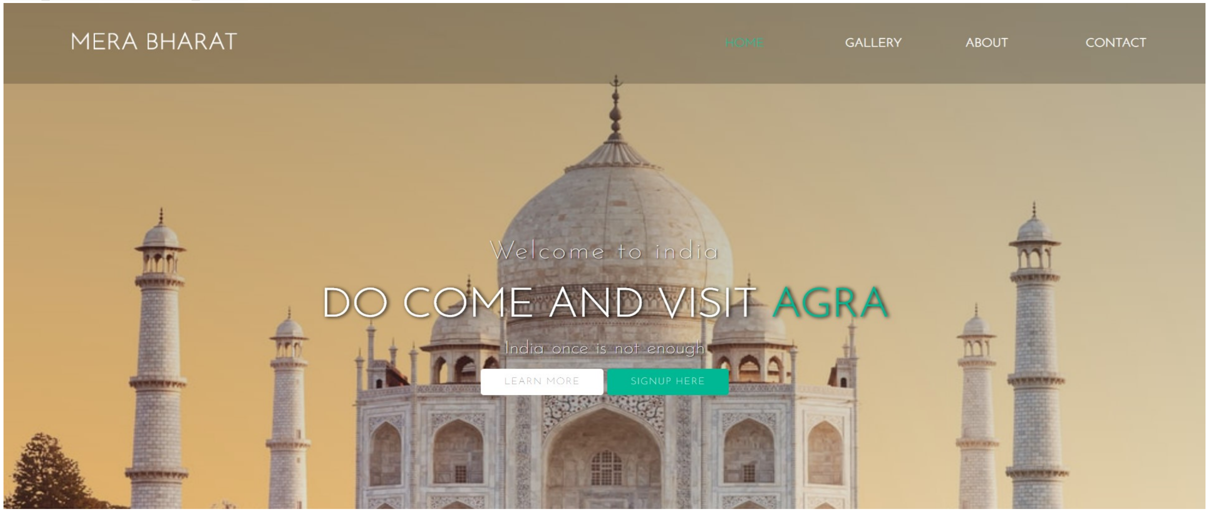
THEORY:

- HTML provides the basic structure of sites, which is enhanced and modified by other technologies like CSS and JavaScript.
- CSS is used to control presentation, formatting, and layout.
- JavaScript is used to control the behavior of different elements.

Tags To be used-

- Animate.css,
- CSS3 Flexbox,
- Animations,
- Transitions property
- HTML5 Semantic property

Expected Output-



For information use reference-

1. **https://www.youtube.com/watch?v=LO4YTml3IAQ**

CONCLUSIONS:

In this Practical we learned objectives of HTML5, CSS and Javascript.

3. Lab Exercise

Exercise No 3: (2 Hours) – 1 Practical

Aim: - Design a Webpage using advance HTML Form tags input–date, time, number, email, HTML5 Header And Footer, spell check and editable areas.

Objectives:

1. Students should able to use Advance HTML Tags.
2. Students should able to differentiate between Basic and Advance HTML Tags.

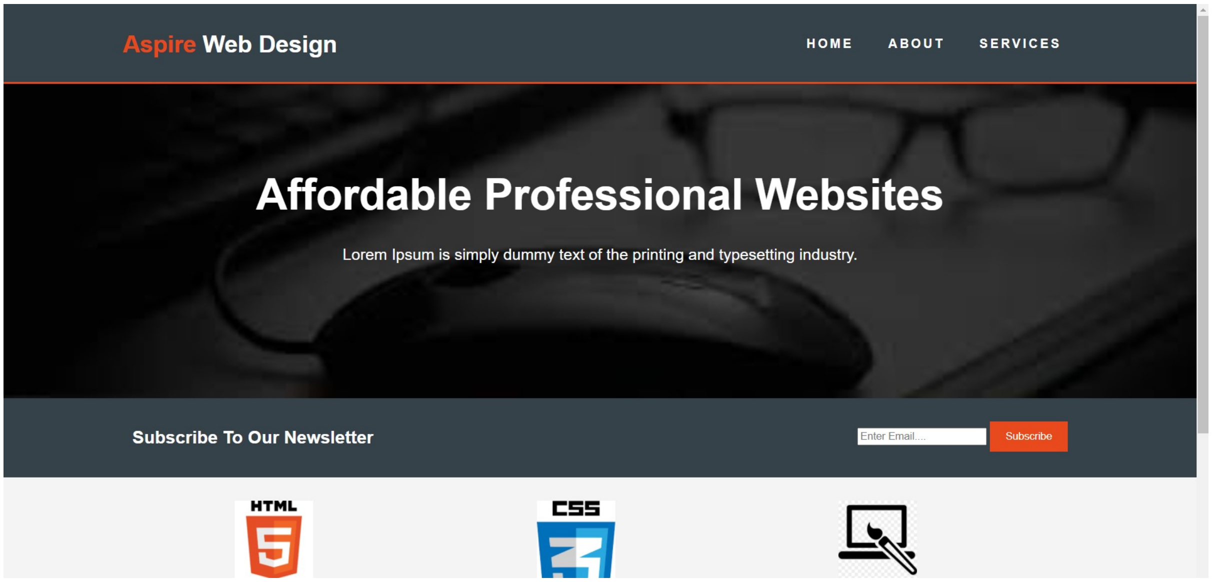
THEORY:

- HTML is made up of a great many elements, a lot of which are overlooked, forgotten or just unknown to many web designers.
- Although with a basic knowledge of HTML you can develop a website, to take advantage of many of the advanced features, and to make pages fully compatible, it is useful to learn these less popular tags.

Tags To be used-

- HTML Form tags input.
- Date,
- Time,
- Number,
- Email,
- HTML5 Header And Footer,
- Spell check and editable areas.
- & all remaining tags.

Expected Output-



Affordable Professional Websites

Lorem Ipsum is simply dummy text of the printing and typesetting industry.

Subscribe To Our Newsletter

For information use reference-

- https://www.w3schools.com/

CONCLUSIONS:

In this Practical we learned how to use advance HTML Tags for webpage creation.

4. Lab Exercise

Exercise No 4: (2 Hours) – 1 Practical

Aim: -

- Design a Webpage Demonstrating Drag and Drop Functionality.
- Implement program demonstrating Local Storage and session storage.

Objectives:

1. Students should able to use Drag and Drop Functionality
2. Students should able to differentiate between Local Storage and session storage.

THEORY:

HTML Drag and Drop-

- Drag and Drop (DnD) is powerful User Interface concept which makes it easy to copy, reorder and deletion of items with the help of mouse clicks. This allows the user to click and hold the mouse button down over an element, drag it to another location, and release the mouse button to drop the element there.

- To achieve drag and drop functionality with traditional HTML4, developers would either have to either have to use complex JavaScript programming or other JavaScript frameworks like jQuery etc.

- Now HTML 5 came up with a Drag and Drop (DnD) API that brings native DnD support to the browser making it much easier to code up.

- HTML 5 DnD is supported by all the major browsers like Chrome, Firefox 3.5 and Safari 4 etc.

HTML Web Storage

- With web storage, web applications can store data locally within the user's browser.

- Before HTML5, application data had to be stored in cookies, included in every server request. Web storage is more secure, and large amounts of data can be stored locally, without affecting website performance.

- Unlike cookies, the storage limit is far larger (at least 5MB) and information is never transferred to the server.

- Web storage is per origin (per domain and protocol). All pages, from one origin, can store and access the same data.

HTML Web Storage Objects

HTML web storage provides two objects for storing data on the client:

- window.localStorage - stores data with no expiration date
- window.sessionStorage - stores data for one session (data is lost when the browser tab is closed)

Tags or Terminologies To be used-

- What to Drag - ondragstart and setData()
- Where to Drop – ondragover
- Do the Drop - ondrop
- window.localStorage
- window.sessionStorage
- `localStorage.setItem`
- `document.getElementById`
- The sessionStorage Object

For information use reference-
https://www.w3schools.com/html/html5_webstorage.asp

CONCLUSIONS:

- In this Practical we learned how to use Drag and Drop Functionality and Local Storage and session storage.

5. Lab Exercise

Exercise No 5: (2 Hours) – 1 Practical

Aim: - Design a Webpage using Basic CSS Tags.
Demonstrate Inline, Internal and External Style sheets using advanced CSS.

Objectives:

1. Students should able to use Advance CSS Property
2. Students should able to differentiate between Basic CSS and Advanced CSS.

THEORY:

- CSS is the language we use to style an HTML document.
- CSS describes how HTML elements should be displayed.
- CSS stands for Cascading Style Sheets
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media
- CSS saves a lot of work. It can control the layout of multiple web pages all at once
- External stylesheets are stored in CSS files

Tags to be used-

- CSS Selectors
- CSS How To
- CSS Comments
- CSS Colors
- CSS Backgrounds
- CSS Borders
- CSS Margins
- CSS Padding
- CSS Height/Width
- CSS Box Model
- CSS Outline
- CSS Text
- CSS Fonts
- CSS Icons
- CSS Links
- CSS Lists
- CSS Tables
- CSS Display
- CSS Max-width
- CSS Position

- CSS Overflow
- CSS Float
- CSS Inline-block
- CSS Align
- CSS Combinators
- CSS Pseudo-class
- CSS Pseudo-element
- CSS Opacity
- CSS Navigation Bar
- CSS Dropdowns
- CSS Image Gallery
- CSS Image Sprites
- CSS Attr Selectors
- CSS Forms
- CSS Counters
- CSS Website Layout
- CSS Units
- CSS Specificity

There are three ways of inserting a style sheet:

- External CSS
- Internal CSS
- Inline CSS

External CSS

- With an external style sheet, you can change the look of an entire website by changing just one file!

- Each HTML page must include a reference to the external style sheet file inside the <link> element, inside the head section.

Internal CSS

- An internal style sheet may be used if one single HTML page has a unique style.

- The internal style is defined inside the <style> element, inside the head section.

Inline CSS

- An inline style may be used to apply a unique style for a single element.

- To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property

Tags to be used-

- CSS Advanced
- CSS Rounded Corners
- CSS Border Images
- CSS Backgrounds
- CSS Colors
- CSS Gradients
- CSS Shadows
- CSS Text Effects
- CSS Web Fonts
- CSS 2D Transforms
- CSS 3D Transforms
- CSS Transitions
- CSS Animations
- CSS Tooltips
- CSS Style Images
- CSS object-fit
- CSS Buttons
- CSS Pagination
- CSS Multiple Columns
- CSS User Interface
- CSS Variables
- CSS Box Sizing
- CSS Media Queries
- CSS MQ Examples
- CSS Flexbox

For information use reference-

- **https://www.w3schools.com/css/**

CONCLUSIONS:

- In this Practical we learned how to use basic CSS tags to style the webpages and how to use Inline, Internal and External Style sheets to style WebPages using advanced CSS.

6. Lab Exercise

Exercise No 6: (2 Hours) – 1 Practical

Aim -Design signup form to validate username, password, and phone numbers etc using Java script.
Write a program to demonstrate Event Handling using JavaScript.(Minimum 3Events)

Objectives:

1. Students should able to use Javascript
2. Students should able to differentiate between basic javascript and advanced javascript.

THEORY:

Data Validation

Data validation is the process of ensuring that user input is clean, correct, and useful.

Typical validation tasks are:

- has the user filled in all required fields?
- has the user entered a valid date?
- has the user entered text in a numeric field?

Most often, the purpose of data validation is to ensure correct user input.

Validation can be defined by many different methods, and deployed in many different ways.

Server-side validation is performed by a web server, after input has been sent to the server.

Client-side validation is performed by a web browser, before input is sent to a web server.

HTML Constraint Validation

HTML5 introduced a new HTML validation concept called **constraint validation**.

HTML constraint validation is based on:

- Constraint validation **HTML Input Attributes**
- Constraint validation **CSS Pseudo Selectors**

- Constraint validation **DOM Properties and Methods**

JavaScript Form Validation

HTML form validation can be done by JavaScript.

If a form field (fname) is empty, this function alerts a message, and returns false, to prevent the form from being submitted:

Example:

```
function validateForm() {
  var x = document.forms["myForm"]["fname"].value;
  if (x == "") {
    alert("Name must be filled out");
    return false;
  }
}
```

Automatic HTML Form Validation

HTML form validation can be performed automatically by the browser:

If a form field (fname) is empty, the required attribute prevents this form from being submitted:

Example:

```
<form action="/action_page.php" method="post">
  <input type="text" name="fname" required>
  <input type="submit" value="Submit">
</form>
```

HTML Events

An HTML event can be something the browser does, or something a user does.

Here are some examples of HTML events:

- An HTML web page has finished loading
- An HTML input field was changed
- An HTML button was clicked

Often, when events happen, you may want to do something.

JavaScript lets you execute code when events are detected.

HTML allows event handler attributes, **with JavaScript code**, to be added to HTML elements.

With single quotes:
<*element event*=**'some JavaScript'**>

With double quotes:
<*element event*=**"some JavaScript"**>

For information use reference-

- **https://www.w3schools.com/css/**

CONCLUSIONS:

In this Practical, Hence, we have study signup form to validate username, password, and phone numbers etc. using Java script and implemented program Event Handling using JavaScript.

7. Lab Exercise

Exercise No 7: (2 Hours) – 1 Practical

Aim - Design a Form using HTML and CSS and accept the data from it and insert it into Database using PHP.

Objectives:

Students should able to use PHP with database.

THEORY:

HTML Forms are required, when you want to collect some data from the site visitor. For example, during user registration you would like to collect information such as name, email address, credit card, etc.

A form will take input from the site visitor and then will post it to a back-end application such as CGI, ASP Script or PHP script etc. The back-end application will perform required processing on the passed data based on defined business logic inside the application.

There are various form elements available like text fields, textarea fields, drop-down menus, radio buttons, checkboxes, etc.

The HTML **<form>** tag is used to create an HTML form and it has following syntax −

```
<form action = "Script URL" method = "GET|POST">
  form elements like input, textarea etc.
</form>
```

HTML Forms are required, when you want to collect some data from the site visitor. For example, during user registration you would like to collect information such as name, email address, credit card, etc.

A form will take input from the site visitor and then will post it to a back-end application such as CGI, ASP Script or PHP script etc. The back-end application will perform required processing on the passed data based on defined business logic inside the application.

There are various form elements available like text fields, textarea fields, drop-down menus, radio buttons, checkboxes, etc.

The HTML **<form>** tag is used to create an HTML form and it has following syntax −

```
<form action = "Script URL" method = "GET|POST">
  form elements like input, textarea etc.
</form>
```

Attributes

Following is the list of attributes for <input> tag for creating text field.

Sr.No	Attribute & Description
1	**type** Indicates the type of input control and for text input control it will be set to **text**.
2	**name** Used to give a name to the control which is sent to the server to be recognized and get the value.
3	**value** This can be used to provide an initial value inside the control.
4	**size** Allows to specify the width of the text-input control in terms of characters.
5	**maxlength** Allows to specify the maximum number of characters a user can enter into the text box.

Password input controls

This is also a single-line text input but it masks the character as soon as a user enters it. They are also created using HTML <input>tag but type attribute is set to **password**.

Multiple-Line Text Input Controls

This is used when the user is required to give details that may be longer than a single sentence. Multi-line input controls are created using HTML <textarea> tag.

Checkbox Control

Checkboxes are used when more than one option is required to be selected. They are also created using HTML <input> tag but type attribute is set to **checkbox.**.

Radio Button Control

Radio buttons are used when out of many options, just one option is required to be selected. They are also created using HTML <input> tag but type attribute is set to **radio**.

Button Controls

There are various ways in HTML to create clickable buttons. You can also create a clickable button using <input>tag by setting its type attribute to **button**. The type attribute can take the following values −

Sr.No	Type & Description
1	**submit** This creates a button that automatically submits a form.
2	**reset** This creates a button that automatically resets form controls to their initial values.
3	**button** This creates a button that is used to trigger a client-side script when the user clicks that button.
4	**image** This creates a clickable button but we can use an image as background of the button.

For information use reference-

- https://www.w3schools.com/

CONCLUSIONS:

In this Practical, Hence, we have studied how to design a Form using HTML and CSS to accept the data from user and insert it into Database using PHP.

8. Lab Exercise

Exercise No 8: (2 Hours) – 1 Practical

Aim - Create a form with a text box asking to enter your favorite city with a submit button when the user enters the city and clicks the submit button another PHP page should be opened displaying “Welcome to the city”.

Objectives:

Students should able to use PHP with all functionality.

THEORY:

The PHP Hypertext Preprocessor (PHP) is a programming language that allows web developers to create dynamic content that interacts with databases. PHP is basically used for developing web-based software applications. This tutorial helps you to build your base with PHP.

PHP is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain. I will list down some of the key advantages of learningPHP:

- PHP is a recursive acronym for "PHP: HypertextPreprocessor".
- PHP is a server-side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites.
- It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQLServer.
- PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the Unix side. The MySQL server, once started, executes even very complex queries with huge result sets in record-settingtime.
- PHP supports a large number of major protocols such as POP3, IMAP, and LDAP. PHP4 added support for Java and distributed object architectures (COM and CORBA), making n-tier development a possibility for the firsttime.

- PHP is forgiving: PHP language tries to be as forgiving aspossible.
- PHP Syntax isC-Like.

Characteristics of PHP

Five important characteristics make PHP's practical nature possible –

- Simplicity
- Efficiency
- Security
- Flexibility

Familiarity

Applications of PHP

As mentioned before, PHP is one of the most widely used language over the web. I'm going to list few of themhere:

- PHP performs system functions, i.e., from files on a system it can create, open, read, write, and closethem.
- PHP can handle forms, i.e., gather data from files, save data to a file, through email you can send data, return data to theuser.
- You add, delete, modify elements within your database throughPHP.
- Access cookies variables and setcookies.
- Using PHP, you can restrict users to access some pages of yourwebsite.
- It can encryptdata.

PHP echo and print Statements

echo and print are more or less the same. They are both used to output data to the screen. The differences are small: echo has no return value while print has a return value of 1 so it can be used in expressions. echo can take multiple parameters (although such usage is rare) while print can take one argument. echo is marginally faster than print.

PHP Data Types

Variables can store data of different types, and different data types can do different things. PHP supports the following data types:

- String
- Integer
- Float (floating point numbers - also calleddouble)
- Boolean
- Array
- Object
- NULL
- Resource

PHP Conditional Statements

Very often when you write code, you want to perform different actions for different conditions. You can use conditional statements in your code to do this.

In PHP we have the following conditional statements:

- if statement - executes some code if one condition istrue
- if...else statement - executes some code if a condition is true and another code if that condition isfalse

- if...elseif...elsestatement-executesdifferentcodesformorethantwoconditions
- switch statement - selects one of many blocks of code to beexecuted

Create an Array in PHP

In PHP, the array () function is used to create an array:

array ();

In PHP, there are three types of arrays:

- **Indexed arrays** - Arrays with a numericindex
- **Associative arrays** - Arrays with namedkeys
- **Multidimensional arrays** - Arrays containing one or morearrays

Create a User Defined Function in PHP

A user-defined function declaration starts with the word function:

Syntax

```
  functionfunctionName () {
codetobeexecuted;
  }
```

<u>Creation of Form in HTML</u>:

The <form> element is a container for different types of input elements, such as: text fields, checkboxes, radio buttons, submit buttons, etc.

The <input> Element

The HTML <input> element is the most used form element.

An <input> element can be displayed in many ways, depending on the type attribute.

For information use reference-

- https://www.w3schools.com/

CONCLUSIONS:

In this Practical, Hence, we have studied how to link Html and PHP pages.

9. Lab Exercise

Exercise No 9: (2 Hours) – 1 Practical

Aim - Change a Content of webpage using AJAX. Perform Different Operations using JQUERY Selectors.

Objectives:

Students should able to use AJAX and JQUERY.

THEORY:

AJAX :

AJAX stands for Asynchronous JavaScript and XML. AJAX is a new technique for creating better, faster, and more interactive web applications with the help of XML, HTML, CSS, and Java Script.

- Ajax uses XHTML for content, CSS for presentation, along with Document Object Model and JavaScript for dynamic content display.
- Conventional web applications transmit information to and from the sever using synchronous requests. It means you fill out a form, hit submit, and get directed to a new page with new information from the server.
- With AJAX, when you hit submit, JavaScript will make a request to the server, interpret the results, and update the current screen. In the purest sense, the user would never know that anything was even transmitted to the server.
- XML is commonly used as the format for receiving server data, although any format, including plain text, can be used.
- AJAX is a web browser technology independent of web server software.
- A user can continue to use the application while the client program requests information from the server in the background.
- Intuitive and natural user interaction. Clicking is not required, mouse movement is a sufficient event trigger.
- Data-driven as opposed to page-driven.
- AJAX is Based on Open Standards

AJAX is based on the following open standards −

- Browser-based presentation using HTML and Cascading Style Sheets (CSS).
- Data is stored in XML format and fetched from the server.
- Behind-the-scenes data fetches using XMLHttpRequest objects in the browser.
- JavaScript to make everything happen.

jQuery

jQuery is a fast and concise JavaScript Library created by John Resig in 2006 with a nice motto: Write less, do more. jQuery simplifies HTML document traversing, event handling, animating, and Ajax interactions for rapid web development. jQuery is a JavaScript toolkit designed to simplify various tasks by writing less code. Here is the list of important core features supported by jQuery −

- DOM manipulation − ThejQuery made it easy to select DOM elements, negotiate them and modifying their content by using cross-browser open source selector engine called Sizzle.
- Event handling − The jQuery offers an elegant way to capture a wide variety of events, such as a user clicking on a link, without the need to clutter the HTML code itself with event handlers.
- AJAX Support − ThejQuery helps you a lot to develop a responsive and featurerich site using AJAX technology.
- Animations − ThejQuery comes with plenty of built-in animation effects which you can use in your websites.
- Lightweight − ThejQuery is very lightweight library - about 19KB in size (Minified and gzipped).
- Cross Browser Support − The jQuery has cross-browser support, and works well in IE 6.0+, FF 2.0+, Safari 3.0+, Chrome and Opera 9.0+
- Latest Technology − ThejQuery supports CSS3 selectors and basic XPath syntax.

JQuerySelector :

A jQuery Selector is a function which makes use of expressions to find out matching elements from a DOM based on the given criteria. Simply you can say, selectors are used to select one or more HTML elements using jQuery. Once an element is selected then we can perform various operations on that selected element.

How to Use Selectors?
The selectors are very useful and would be required at every step while using jQuery. They get the exact element that you want from your HTML document.

Following table lists down few basic selectors and explains them with examples.

Sr.No. Selector & Description

1 Name

Selects all elements which match with the given element Name.

2 #ID

Selects a single element which matches with the given ID.

3 .Class

Selects all elements which match with the given Class.

4 Universal (*)

Selects all elements available in a DOM.

5 Multiple Elements E, F, G

Selects the combined results of all the specified selectors E, F or G.

Selector &Description :
1 $('*')
This selector selects all elements in the document.
2 $("p > *")
This selector selects all elements that are children of a paragraph element.
3 $("#specialID")
This selector function gets the element with id="specialID".
4 $(".specialClass")
This selector gets all the elements that have the class of specialClass.
5 $("li:not(.myclass)")
Selects all elements matched by <li> that do not have class = "myclass".
6 $("a#specialID.specialClass")
This selector matches links with an id of specialID and a class of specialClass.
7 $("p a.specialClass")
This selector matches links with a class of specialClass declared within <p> elements.
8 $("ulli:first")
This selector gets only the first <li> element of the <ul>.
9 $("#container p")
Selects all elements matched by <p> that are descendants of an element that has an id of container.
10 $("li >ul")
Selects all elements matched by <ul> that are children of an element matched by <li>
11 $("strong + em")
Selects all elements matched by <em> that immediately follow a sibling element matched by <strong>.
12 $("p ~ ul")
Selects all elements matched by <ul> that follow a sibling element matched by <p>.
13 $("code, em, strong")
Selects all elements matched by <code> or <em> or <strong>.
14 $("p strong, .myclass")
Selects all elements matched by <strong> that are descendants of an element matched by <p> as well as all elements that have a class of myclass.
15 $(":empty")
Selects all elements that have no children.
16 $("p:empty")
Selects all elements matched by <p> that have no children.
17 $("div[p]")
Selects all elements matched by <div> that contain an element matched by <p>.
18 $("p[.myclass]")
Selects all elements matched by <p> that contain an element with a class of myclass.
19 $("a[@rel]")
Selects all elements matched by <a> that have a rel attribute.
20 $("input[@name = myname]")
Selects all elements matched by <input> that have a name value exactly equal to myname.
21 $("input[@name^=myname]")

Selects all elements matched by <input> that have a name value beginning with myname.
22 $("a[@rel$=self]")
Selects all elements matched by <a> that have rel attribute value ending with self.
23 $("a[@href*=domain.com]")
Selects all elements matched by <a> that have anhref value containing domain.com.
24 $("li:even")
Selects all elements matched by <li> that have an even index value.
25 $("tr:odd")
Selects all elements matched by <tr> that have an odd index value.
26 $("li:first")
Selects the first <li> element.
27 $("li:last")
Selects the last <li> element.
28 $("li:visible")
Selects all elements matched by <li> that are visible.
29 $("li:hidden")
Selects all elements matched by <li> that are hidden.
30 $(":radio")
Selects all radio buttons in the form.
31 $(":checked")
Selects all checked box in the form.
32 $(":input")
Selects only form elements (input, select, textarea, button).
33 $(":text")
Selects only text elements (input[type = text]).
34 $("li:eq(2)")
Selects the third <li> element.
35 $("li:eq(4)")
Selects the fifth <li> element.
36 $("li:lt(2)")
Selects all elements matched by <li> element before the third one; in other words, the first two <li> elements.
37 $("p:lt(3)")
selects all elements matched by <p> elements before the fourth one; in other words the first three
<p> elements.
38 $("li:gt(1)")
Selects all elements matched by <li> after the second one.
39 $("p:gt(2)")
Selects all elements matched by <p> after the third one.
40 $("div/p")
Selects all elements matched by <p> that are children of an element matched by <div>.
41 $("div//code")
Selects all elements matched by <code>that are descendants of an element matched by <div>.
42 $("//p//a")
Selects all elements matched by <a> that are descendants of an element matched by <p>
43 $("li:first-child")

Selects all elements matched by <li> that are the first child of their parent.
44 $("li:last-child")
Selects all elements matched by <li> that are the last child of their parent.
45 $(":parent")
Selects all elements that are the parent of another element, including text.
46 $("li:contains(second)")
Selects all elements matched by <li> that contain the text second.

For information use reference-

- https://www.w3schools.com/

CONCLUSIONS:

In this Practical, Hence, we have studied how to use AJAX and jQuery for implementation.

.

10. Lab Exercise

Exercise No 10: (2 Hours) – 1 Practical

Aim – Mini-Project

Objectives:
Students should able to use all terminologies of Full Stack development through Mini project.

Code-

Output/Screenshots-