



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 : Second Year

Semester : III

Course Code :

Course Title : HTML & Java Script

Prepared By

S.A.Kharat

Assistant Professor

Department of Computer Science & Engineering

FOREWORD

It is my great pleasure to present this laboratory manual for **Second year** engineering students for the subject of HTML & Java Script.

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 Second year students of Computer Science & Engineering in the subject of Introduction to Web Programming. This manual typically contains practical/Lab Sessions related HTML and JavaScript covering various aspects related the subject to enhanced understanding.

HTML and JavaScript provides students the idea of Design Web pages. It also helps to understands the concept of JavaScript programming in designing a webpage

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.S.A.Kharat
Subject Teacher

Dr. Vijaya Musande
HOD

LIST OF EXPERIMENTS

Course Code: CS1037

Course Title: HTML & JAVA SCRIPT

S.No	Name of the Experiment	Page No
1.	Prepare a survey document of ten website which you like and dislike with various reasons. (Prerequisite)	
2.	Introduction to basic HTML elements	
3.	Use table tag to format web page. Also create the Time Table of your class using table tag.	
4.	Create your profile page i.e. educational details, Hobbies, Achievement, My Ideals etc.	
5.	Create Style sheet to set formatting for text tags and embed that style sheet on web pages created for your site.	
6.	Design a web page and embed various multimedia features in the page.	

7.	Design signup form to validate username, password, and phone numbers etc using JavaScript.	
8.	Write a JavaScript program to determine whether a given year is a leap year in the Gregorian calendar.	
9.	Write a JavaScript program to convert temperatures to and from celsius, Fahrenheit.	
10.	Installation of Wordpress and designing the wordpress site.	
11.	Introduction to Dreamweaver and setting up site using Dreamweaver.	
12.	Submission of Website with Report.	

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.



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: Create and Link web page documents.

LO-2: Learn and Understand the different CSS.

LO-3: Create custom styles and use it to style the web.

LO-4: Work with web development tools like Wordpress & Dreamweaver.

LO-5: Implement decision statements in Javascript.

LO-6: Create a DHTML web page using HTML & Javascript.

1. Lab Exercise

Exercise No 1: (2 Hours) – 1 Practical

Aim: - Introduction to basic HTML elements.

Tools: Notepad Editor and Web Browser like Internet Explorer or Mozilla Firefox.

Standard Procedure for Creating and View an HTML document?

- Use a text editor such as Notepad to write the document.
- Save the file as filename.html on a PC. This is called the Document Source.
- Open the file that you have saved in any browser Off-Line.
- Your HTML page should now appear just like any other Web page in browser.
- You may now switch back and forth between the Source and the HTML Document
- switch to Notepad with the Document Source
- make changes .
- save the document again.
- switch back to browser .
- click on RELOAD and view the new HTML Document .
- switch to Notepad with the Document Source.

THEORY:

1. What is HTML?

- A series of tags that are integrated into a text document.
- A series of tags that are integrated into a text document.
- These look like: `<code>formatted text</code>` o `<code>` begins the formatting tag. o `</code>` ends the formatting tag.
- These tags are then read by a Browser, which translates the tags into the formatting that they represent

2. What are Tags?

- HTML tags are used to mark-up HTML elements.
- HTML tags are surrounded by the two characters `<` and `>`.
- The surrounding characters are called angle brackets .
- HTML tags normally come in pairs like `` and ``.
- The first tag in a pair is the start tag; the second tag is the end tag.
- The text between the start and end tags is the element content.
- HTML tags are not case sensitive; `` means the same as ``.

3. Structure Tags In HTML

HTML Tag

```
<HTML></HTML>
```

These tags begin and end an HTML document.

HEAD Tag

```
<HEAD></HEAD>
```

These tags are in the beginning of the document. Important information is stored in-between these tags including: title, meta-data, styles, and programming scripts

TITLE Tag

```
<TITLE></TITLE>
```

These tags are in-between the HEAD tags and contain the text that appears in the title of the Web page.

BODY Tag

```
<BODY></BODY>
```

As you may have guessed, the BODY tags contain all the text in the body of the document.

4. Block Level tags

4.1 HTML Headings –

Headings are defined with the<h1> to<h6> tags.

Where <h1> -Defines the largest headings.

<h6> -Defines the smallest headings.

4.2HTML Paragraphs -

Paragraphs are defined with the<p> tag.

4.3 HTML Line Breaks -

Use the
 tag if you want a line break (a new line) without starting a new paragraph.

4.4 Horizontal Rule

The <hr> element is used for horizontal rules that act as dividers between sections

5. HTML Text Formatting Tags

Tag	Description
-----	-------------

	Defines bold Text
<big>	Defines big text.
	Defines emphasized text
<i>	Defines Italic text
<small>	Defines italic text.
	Defines Strong text
<sub>	Defines subscripted text
<sup>	Defines superscripted text
<ins>	Defines inserted text
	Defines deleted text

CONCLUSIONS:

In this way we can use different basic tags of HTML for creating webpage displaying information.

2. Lab Exercise

Exercise No 2: (2 Hours) – 1 Practical

Aim: - Design a timetable and display it in tabular format.

Tools: Notepad Editor and Web Browser like Internet Explorer or Mozilla Firefox.

THEORY:

HTML Tables:

- Tables are defined with the <table> tag.
- A table is divided into rows (with the <tr> tag),
- Each row is divided into data cells (with the <td> tag).
- td stands for "table data," and holds the content of a data cell.
- A <td> tag can contain text, links, images, lists, forms, other tables, etc.

HTML Tables and the Border Attribute

If you do not specify a border attribute, the table will be displayed without borders. Sometimes this can be useful, but most of the time, we want the borders to show.

To Create Table Header

```
<html>
<body>
<h4>Table headers:</h4>
<table border="1">
<tr>
<th>Name</th>
<th>Telephone</th>
<th>Telephone</th>
</tr>
<tr>
<td>Bill Gates</td>
<td>555 77 854</td>
<td>555 77 855</td>
</tr>
</table>
</body>
</html>
```

Output

Name	Telephone	Telephone
Bill Gates	555 77 854	555 77 855

How to add a caption to a table.

`<Caption>Monthly savings</caption>`

Table Cells then spans more than one row/column

For Example-

```
<html>
<body>
<h4>Cell that spans two columns:</h4>
<table border="1">
<tr>
<th>Name</th>
<th colspan="2">Telephone</th>
</tr>
<tr>
<td>Bill Gates</td>
<td>555 77 854</td>
<td>555 77 855</td>
</tr>
</table>
<h4>Cell that spans two rows:</h4>
<table border="1">
<tr> <th>First Name:</th>
<td>Bill Gates</td>
</tr>
<tr>
<th rowspan="2">Telephone:</th>
<td>555 77 854</td>
</tr>
<tr>
<td>555 77 855
</td>
</tr>
</table>
</body>
</html>
```

Output-

Cell that spans two columns:

Name	Telephone	
Bill Gates	555 77 854	555 77 855

Cell that spans two rows:

First Name:	Bill Gates
Telephone:	555 77 854
	555 77 855

CONCLUSIONS:

With the help of given information about the Table and Table tags we can design a time table in tabular format.

3. Lab Exercise

Exercise No 3: (2 Hours) – 1 Practical

Aim: -Create your profile page i.e. educational details, Hobbies, Achievement, My Ideals etc.

Tools: Notepad Editor and Web Browser like Internet Explorer or Mozilla Firefox.

STANDARD PROCEDURE:

- ♣ Step 1: Create **Educational_Details.html** page.
- ♣ Step 2: Create **Hobbies.html** page.
- ♣ Step 3: Create **Achievements.html** page.
- ♣ Step 4: Create **My_Ideals.html** page.
- ♣ Step 5: Create links of above four html pages in **Link.html** page.

THEORY:

HTML uses the <a> anchor tag to create a link to another document or web page.

The Anchor Tag and the href Attribute

- ♣ An anchor can point to any resource on the Web: an HTML page, an image, a sound file, a movie, etc. The syntax of creating an anchor:
- ♣ `Text to be displayed`
- ♣ The <a> tag is used to create an anchor to link from, the href attribute is used to tell the address of the document or page we are linking to, and the words between the open and close of the anchor tag will be displayed as a hyperlink.

The Target Attribute

- ♣ With the target attribute, you can define **where** the linked document will be opened. By default, the link will open in the current window.
- ♣ The code below will open the document in a new browser window.
`Visit google!`

Email Links

- To create an email link, you will use mailto: plus your email address. Here is a link to ACC's Help Desk:

`Email Help Desk`

- To add a subject for the email message, you would add ?subject= after the email address. For example:

`Email Help Desk`

CONCLUSIONS:

With the help of given procedure and information about the Links we can write HTML code for creating Hyperlinks in html page.

4. Lab Exercise

Exercise No 4: (2 Hours) – 1 Practical

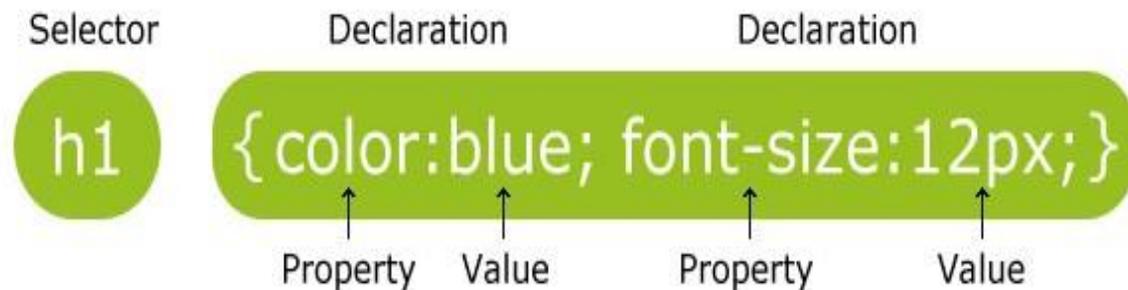
Aim: - Create Style sheet to set formatting for text tags and embed that style sheet on web pages created for your site.

Tools: Notepad Editor and Web Browser like Internet Explorer or Mozilla Firefox.

THEORY:

CSS Syntax

A CSS rule has two main parts: a selector, and one or more declarations:



The selector is normally the HTML element you want to style.

Each declaration consists of a property and a value.

The property is the style attribute you want to change. Each property has a value.

CSS Example

A CSS declaration always ends with a semicolon, and declaration groups are surrounded by curly brackets:

```
p {color:red;text-align:center;}
```

Types of CSS-

1. **Inline CSS**
2. Here “Style” Attribute is used.

For Example-

```
<body style="background-color: red;">
```

2. Internal CSS

Here HTML tag `<style>` is used.

For

Example-

```
<html>
<head>
<title>Example</title> <style
type="text/css"> body
{ background-color:
#FF0000;} p {background-
color: yellow;}
</style>
</head>
<body>
<p>This is a red page</p>
</body>
</html>
```

3. External CSS

Here we need to create two separate file `.css` file and `.html` file.

For Example-

HTML file consisting of “`mystyle.css`”

```
<html>
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css" />
</head>
<body>
<p> paragraph</p>
<hr>
</body>
</html>
```

CSS Code- CSS files saved as “`mystyle.css`”

```
<html>
<head>
hr {color:sienna;} p {margin-left:20px;} body
{background-image:url("images/Blue hills.jpg");}
</head>
</html>
```

CONCLUSIONS:

With the help of given information about CSS we can create webpage using css.

5. Lab Exercise

Exercise No 5: (2 Hours) – 1 Practical

Aim: - Design a web page and embed various multimedia features in the page.

Tools: Notepad Editor and Web Browser like Internet Explorer or Mozilla Firefox.

THEORY:

What is Multimedia?

Multimedia comes in many different formats. It can be almost anything you can hear or see.

Examples: Images, music, sound, videos, records, films, animations, and more.

Web pages often contain multimedia elements of different types and formats.

The HTML <video> Element

To show a video in HTML, use the `<video>` element:

HTML5 defines DOM methods, properties, and events for the `<video>` element.

This allows you to load, play, and pause videos, as well as setting duration and volume.

There are also DOM events that can notify you when a video begins to play, is paused, etc.

Tag	Description
-----	-------------

<code><video></code>	Defines a video or movie
----------------------------	--------------------------

<code><source></code> <code><audio></code>	Defines multiple media resources for media elements, such as <code><video></code> and <code><audio></code>
---	--

<code><track></code>	Defines text tracks in media players.
----------------------------	---------------------------------------

The HTML <audio> Element

To play an audio file in HTML, use the `<audio>` element:

Tag	Description
<code><audio></code>	Defines sound content
<code><source></code> <code><audio></code>	Defines multiple media resources for media elements, such as <code><video></code> and <code><audio></code>

HTML Audio - Media Types

File Format	Media Type
MP3	audio/mpeg
OGG	audio/ogg
WAV	audio/wav

CONCLUSIONS:

With the help of given procedure and information we can embed multimedia inside a webpage.

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

Tools: Notepad Editor and Web Browser like Internet Explorer or Mozilla Firefox.

THEORY:

Introduction to JavaScript

Variables in JavaScript

Declaring JavaScript variables

First, you need to declare your variables. You do this using the `var` keyword. You can declare one variable at a time or more than one. You can also assign values to the variables at the time you declare them.

```
// declaring one javascript variable
var firstName;

// declaring multiple javascript variables
var firstName, lastName;

// declaring and assigning one javascript variable
var firstName = 'Homer';

// declaring and assigning multiple javascript variables
var firstName = 'Homer', lastName = 'Simpson';
```

When you write JavaScript function, you will you will need to determine when it will run, Often this will be when a user does something like **submit** a form, Validation prompt should appears.

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 Example

In this example, we are going to validate the name and password. The name can't be empty and password can't be less than 6 characters long.

Here, we are validating the form on form submit. The user will not be forwarded to the next page until given values are correct.

```
<script>
```

```
function validateform(){
```

```
var name=document.myform.name.value;
```

```
var password=document.myform.password.value;
```

```
if (name==null || name==""){
    alert("Name can't be blank");
    return false;
}else if(password.length<6){
    alert("Password must be at least 6 characters long.");
    return false;
}
}
}
</script>
<body>
<form name="myform" method="post" action="abc.jsp" onsubmit="return validateform()" >
Name: <input type="text" name="name"><br/>
Password: <input type="password" name="password"><br/>
<input type="submit" value="register">
</form>
```

JavaScript Retype Password Validation

```
<script type="text/javascript">
function matchpass(){
    var firstpassword=document.f1.password.value;
    var secondpassword=document.f1.password2.value;

    if(firstpassword==secondpassword){
        return true;
```

```
}  
else{  
alert("password must be same!");  
return false;  
}  
}  
</script>  
<form name="f1" action="register.jsp" onsubmit="return matchpass()">  
Password:<input type="password" name="password" /><br/>  
Re-enter Password:<input type="password" name="password2"/><br/>  
<input type="submit">  
</form>
```

CONCLUSIONS:

With the help of given information about the use of Events and Functions, We can Design a Signup form with all validations

7. Lab Exercise

Exercise No 7: (2 Hours) – 1 Practical

Aim: - Write a JavaScript program to determine whether a given year is a leap year in the Gregorian calendar.

Tools: Notepad Editor and Web Browser like Internet Explorer or Mozilla Firefox.

THEORY:

Introduction to JavaScript

JavaScript is not a compiled language, but it is a translated language. The JavaScript Translator (embedded in the browser) is responsible for translating the JavaScript code for the web browser.

A leap year is a year containing one additional day added to keep the calendar year synchronized with the astronomical or seasonal year. Because seasons and astronomical events do not repeat in a whole number of days, calendars that have the same number of days in each year drift over time with respect to the event that the year is supposed to track. By inserting an additional day or month into the year, the drift can be corrected. A year that is not a leap year is called a common year.

Every year that is exactly divisible by four is a leap year, except for years that are exactly divisible by 100, but these centurial years are leap years if they are exactly divisible by 400. For example, the years 1700, 1800, and 1900 are not leap years, but the year 2000 is.

To determine whether a year is a leap year, follow these steps:

Step-1 : If the year is evenly divisible by 4, go to step 2. Otherwise, go to step 5.

Step-2 : If the year is evenly divisible by 100, go to step 3. Otherwise, go to step 4.

Step-3 : If the year is evenly divisible by 400, go to step 4. Otherwise, go to step 5.

Step-4 : The year is a leap year (it has 366 days).

Step-5 : The year is not a leap year (it has 365 days).

Sample Solution:

HTML Code:

```
<!DOCTYPE html>

<html>

<head>

<meta charset=utf-8 />

<title>Find Leap Year</title>

</head>

<body>

</body>

</html>
```

JavaScript Code:

```
function leapyear(year)
{
return (year % 100 === 0) ? (year % 400 === 0) : (year % 4 === 0);
}

console.log(leapyear(2016));

console.log(leapyear(2000));
```

```
console.log(leapyear(1700));
```

```
console.log(leapyear(1800));
```

```
console.log(leapyear(100));
```

CONCLUSIONS:

With the help of given procedure and information we have written javascript to find given year is leap or not.

8. Lab Exercise

Exercise No 8: (2 Hours) – 1 Practical

Aim: - Write a JavaScript program to convert temperatures to and from Celsius, Fahrenheit

Tools: Notepad Editor and Web Browser like Internet Explorer or Mozilla Firefox.

THEORY:

JavaScript: Fahrenheit and Centigrade Temperatures:

Fahrenheit and centigrade are two temperature scales in use today. The Fahrenheit scale was developed by the German physicist Daniel Gabriel Fahrenheit. In the Fahrenheit scale, water freezes at 32 degrees and boils at 212 degrees.

The centigrade scale, which is also called the Celsius scale, was developed by Swedish astronomer Andres Celsius. In the centigrade scale, water freezes at 0 degrees and boils at 100 degrees. The centigrade to Fahrenheit conversion formula is:

$$C = (5/9) * (F - 32)$$

where F is the Fahrenheit temperature. You can also use this Web page to convert Fahrenheit temperatures to centigrade

Sample Solution:

HTML Code:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>Write a JavaScript program to convert temperatures to and from celsius,
fahrenheit</title>
</head>
```

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

JavaScript Code:

```
function cToF(celsius)
{
  var cTemp = celsius;
  var cToFahr = cTemp * 9 / 5 + 32;
  var message = cTemp + '\xB0C is ' + cToFahr + ' \xB0F.';
  console.log(message);
}

function fToC(fahrenheit)
{
  var fTemp = fahrenheit;
  var fToCel = (fTemp - 32) * 5 / 9;
  var message = fTemp + '\xB0F is ' + fToCel + ' \xB0C.';
  console.log(message);
}

cToF(60);
fToC(45);
```

CONCLUSIONS:

For an exact conversion (Fahrenheit to Celsius / Celsius to Fahrenheit) the following formulas can be applied :

Fahrenheit to Celsius : $(^{\circ}\text{F} - 32) \div 1.8 = ^{\circ}\text{C}$

Celsius to Fahrenheit : $(^{\circ}\text{C} \times 1.8) + 32 = ^{\circ}\text{F}$

An approximate conversion between degrees Celsius and degrees Fahrenheit is as follows :

Fahrenheit to Celsius : Subtract 30 and halve the resulting number.

Celsius to Fahrenheit : Double the number and add 30.

9. Lab Exercise

Exercise No 9: (2 Hours) – 1 Practical

Aim: - Installation of Wordpress and designing the wordpress site.

Tools: Notepad Editor and Web Browser like Internet Explorer or Mozilla Firefox.

THEORY:

Installing & Configuring WordPress:

If your new to the WordPress game and have never setup a site before then these steps will walk you through it. These steps assume you are starting from scratch and all you have is an idea lodged in your head or written down on paper.

Basic Steps to creating a WordPress based site:

1. Register domain name
2. Register hosting
3. Install a clean version of WordPress
4. Point name servers to hosting
5. Change password for WordPress login
6. Find and upload a theme
7. Install and configure plugins
8. Customize theme
9. Create Pages
10. Setup menus
11. Configure SEO using a plugin
12. Go live with your site

CONCLUSIONS:

With the help of given procedure and information we can install wordpress and setup the wordpress site.

10. Lab Exercise

Exercise No 10: (2 Hours) – 1 Practical

Aim: - Introduction to Dreamweaver and setting up site using Dreamweaver.

Tools: Notepad Editor and Web Browser like Internet Explorer or Mozilla Firefox.

THEORY:

Introduction to Dreamweaver

Dreamweaver – what is it? Adobe Dreamweaver is a software application that allows you to create and develop Web sites. Dreamweaver is considered WYSIWYG (What You See Is What You Get), meaning that when you format your Web page, you see the results of the formatting instead of the mark-ups that are used for formatting. However, Dreamweaver allows you to hand code HTML as well. Dreamweaver also supports CSS and JavaScript as well as other languages including ASP and PHP. Dreamweaver makes it easy to upload your entire Web site to a Web server. You can also preview your site locally. Dreamweaver also lets you create templates for your Web site that you can use again and again by modifying certain unrestricted areas within the template. Then if you want to change one particular part of your Web site (the logo changes, a main link changes), you only have to modify the template for the changes to propagate throughout your Web site.

In this lab we will be getting a basic introduction to this software package.

Using Dreamweaver:

1. Start Dreamweaver. You should see a startup screen.
2. Create a New Site

Making your first Web Page:

3. You should now see a blank Web page in the Document Window.
4. Under the File Menu (top left of Dreamweaver work area), choose Save.
5. In the Save As box, find your Web site folder and save your file as index.html
6. Click Save. Congratulations! You've created your first Web page using Dreamweaver.

Unfortunately, it's pretty boring at the moment.

7. Give your Web Page a Title
8. Adding Some Content
9. Playing with Font Properties
10. Alignment etc.

Setting Up a New Site in Dreamweaver:

Dreamweaver's strength lies in its powerful site creation and management tools. You can use the software to create everything from individual pages to complete websites. The pages you create within your site can share similar topics, a cohesive design, or a common purpose. And, once your Dreamweaver site is complete, you can efficiently manage and distribute it from within the program.

Starting up

Before starting, make sure that your tools and panels are consistent by resetting your workspace.

Creating a new site

In Dreamweaver, the term site refers to the local and remote storage locations where the files that make up a website are stored. A site can also include a testing server location for processing dynamic pages. To take full advantage of Dreamweaver's features, you should always start by creating a site. Dreamweaver CC can also be used to setup and manage a Business Catalyst Site. Business Catalyst is a part of the Creative Cloud service that Adobe provides (for a fee) that allows you to connect your website with analytic software and other features.

The easiest way to create a standard new site in Dreamweaver is to use the Site Setup dialog box. One way to access this dialog box is by choosing Site > New Site from the menu bar.

In this lesson, you begin by using the Site Setup dialog box to accomplish the following tasks:

- Define the site
- Name the site

- Define the local root folder
- Set up a remote folder
- Explore advanced settings
- Save the site

By default, the Site Setup dialog box opens with the Site Settings available. The options available here will help guide you through the essentials of defining your site. The Servers, Version Control, and Advanced Settings options allow you to set up local, remote, and testing servers directly.

1. Launch Dreamweaver CC, if it is not already open, then choose Site > New Site. First, you have to name the site. In the Site Name text field, type Organic Utopia.

Type the Site Name into the text field.

Next, you need to set up a local root folder, which is where Dreamweaver stores the files with which you're currently working. The Local Site Folder field allows you to enter information regarding where you'll be working with your files during development.

2. Click the Folder icon to the right of the Local Site Folder text field to navigate to any pre-existing files.

If you did not click on the folder icon and just clicked Save, Dreamweaver would simply create a new folder on your system where you could begin to create new pages in your site. In this case, you will be pointing to a preexisting folder that already has files within it.

It is important to distinguish between adding a new site (which is what you are doing now) and creating a new site from scratch. In both cases, the important part is that Dreamweaver knows where this folder is on your system. This folder is known as the root folder and will always contain the content that will eventually be your website.

3. Navigate to your desktop and locate the dw02lessons folder you copied to your desktop earlier.

4. Select the dw02lessons folder. On the Windows platform, click Open to open this folder, then press Select (Windows). On the Mac OS platform, click Choose to choose this as your local root folder. The field now shows the path to your newly defined local root folder.

At this point, you have done the minimum amount of steps required to begin working on a site. Now you will take a look at some of the optional features within the Site Setup process.

5. Click the Servers tab. This section allows you to define the remote server where your website will end up being hosted. Take a moment to read the heading in the dialog box. Note that it says you do not need to fill in this information to begin creating a website. It is only necessary if you are connecting to the Web.

6. Click the + button and the Basic site settings window appears. Here there are fields for Server Name, Connect Using, FTP Address, Username and Password, along with other options. These settings allow you to choose both a destination and a method (FTP being the most common) for Dreamweaver to use to transfer files.

Set up access to your remote folder.

As noted earlier, this is an optional step, and you do not have to define your remote folder at this stage. Dreamweaver allows you to define your remote folder at a later time, such as when you're ready to upload.

7. Click the Advanced tab. Click the Server Model menu in the Testing Server section. Here there are choices for different scripting languages such as PHP and ASP pages. If you are an advanced user, this is where you would set up the connections to your testing server.

Again, you won't be making any changes here, so click Cancel.

8. Click the Version Control option on the left to access Subversion settings. Subversion, a VCS (or version control system), keeps track of changes made to files, enabling users to track changes and return to previous versions of any file. For this exercise, make sure the Access pull-down in this window is set to None, as you won't be using Subversion.

You've now completed the site setup process using basic settings.