

IT::Web Technology

Credits: 04

Teaching Scheme Theory:2 Hours/Week

Tut: 1 Hours/Week

Lab: 2 Hours/Week

Course Prerequisites Computer Programming, Database Management Systems, Computer Network

Course Outcomes:

The student will be able to –

1. To learn the fundamental tags of HTML5 and CSS.
2. To obtain knowledge of JavaScript as client-side technology in web development.
3. To understand use of jQuery in dynamic website designing.
4. To acquire skills of server-side technologies such as PHP in web development.
5. To build user interface or front end of complex websites using react.
6. To study building the backend of web application using NodeJS framework.

Programme Outcomes (PO)

PO1	<u>GA: 1</u> Engineering Knowledge	PO7	<u>GA: 7:</u> Environment and sustainability
PO2	<u>GA: 2</u> Problem Analysis	PO8	<u>GA: 8 :</u> Ethics
PO3	<u>GA: 3:</u> Design/ Development of solution	PO9	<u>GA: 9:</u> Individual and Team Work
PO4	<u>GA: 4:</u> Conduct Investigation of ComplexProblems	PO10	<u>GA: 10:</u> Communication
PO5	<u>GA: 5:</u> Modern Tool Usage	PO11	<u>GA: 11 :</u> Project Management and Finance
PO6	<u>GA: 6 :</u> The Engineer and Society	PO12	<u>GA: 12:</u> Lifelong Learning

Programme Specific Outcomes (PSO)

PSO	PSO Focus	PSO Statement
PSO1	Engineering Foundation	Comprehend and apply computing theory principles, data structures, algorithms and programming paradigms pertaining to application domains
PSO2		Use of mathematical, statistical approaches for data handling and modeling
PSO3	Engineering Application	Create, enhance, and deliver application softwares along with effective use of project management skills.
PSO4		Apply computational techniques and development methodologies by analyzing and designing solution for real world problems.

Syllabus

SECTION-I

- **Introduction:** Introduction to web technology, Internet and WWW, web site planning and design issues, HTML5: structure of html document, commenting, formatting tags, list tags, hyperlink tags, image, table tags, frame tags, form tags, CSS, Bootstrap, JSON (6 Hrs)
- **Client Side Technologies:** JavaScript: Overview of JavaScript, Data types, Control Structures, Arrays, Functions and Scopes, Objects in JS, Form validation, DOM: Introduction, DOM levels, DOM Objects, their properties and methods, Manipulating DOM (6 Hrs)
- **JQuery:** Introduction, Loading JQuery, selecting elements, changing styles, creating elements, appending elements, removing elements, handling events. (2 Hrs)

SECTION-II

Server Side Technologies: Introduction, Features, PHP syntax, Control structures, Functions, Arrays, String, Form Handling, File Handling, Session and Cookies, Error Handling, MySQL with PHP (6 Hrs)

React: Introduction, Architecture, Components, JSX, Class, State, Props, Events, Render function, Forms, Lists, Router, Flux. (4 Hrs)

NodeJS: Introduction, Installation of Node JS, Node JS Modules, Node Package Manager (npm), Creating Web server, File System, Express JS, Serving Static Resources, Database connectivity. (4 Hrs)

List of Practicals: (Any Six)

- 1) Installation, configuration and understanding working of XAMPP server for local host.
- 2) Design and implement a web page to demonstrate the use of different HTML tags.
- 3) Design and develop a web page demonstrating the use of CSS tags.
- 4) Design and develop a HTML form for student registration.
- 5) Write a code for validation of student registration form using JavaScript.
- 6) Design and develop a web page to demonstrate various methods of objects in JavaScript like Array, String, Math, Date.
- 7) Design and develop a web page demonstrating various effects using jQuery.
- 8) Write a PHP program to create a simple calculator that can accept two numbers and perform operations like add, subtract, multiplication and divide. Validate input values and prompt/alerts for invalid values.
- 9) Design a dynamic web application using PHP and MYSQL as back-end to perform insert, delete, view and update operation.
- 10) Design a web page demonstrating file handling operations like open, read, write, append copy, move, delete and rename using nodeJS.
- 11) Design and implement simple website using React.

List of Projects:

1. Student Registration System
2. Tours and Travel System
3. Canteen Food Ordering and Management System.
4. Online Personal Counseling
5. Online Recruitment System
6. Farming Assistant System
7. Hospital management System
8. Hostel Management System
9. Online Event Management
10. Online Bus/Railway/Airways Booking System
11. Online Banking System

List of Tutorials: (Any Three)

- 1) Learn various HTML tags
- 2) Use of Bootstrap to design a web page
- 3) Understand use of JavaScript in form validation
- 4) Study of Cookies in PHP
- 5) Study of Session in PHP
- 6) Study of different ways of working with PHP and MySQL
- 7) Express Framework
- 8) Laravel Framework
- 9) RESTFul API
- 10) React Hooks
- 11) Node JS and relational databases
- 12) Node JS and NoSQL databases

List of Course Seminar Topics:

1. Bootstrap
2. Spring Framework
3. Joomla
4. Sass
5. Java Servlets
6. Object Oriented PHP
7. Angular JS
8. VueJS
9. Django
10. Laravel

List of Course Group Discussion Topics:

1. Web Services
2. Client Side Frameworks
3. Server Side Frameworks
4. Relational and NoSQL Databases
5. AJAX
6. Client Side Technologies
7. Server Side Technologies
8. Template Engine
9. Progressive Web Apps
10. Markup Languages

List of Home Assignments:

- **Design:**

1. Design, Develop and Deploy social web applications using Bootstrap.
2. Design, Develop and Deploy web applications using CMS.
3. Design, Develop and Deploy web application for Electricity Billing System
4. Design, Develop and Deploy web application for department
5. Design, Develop and Deploy web application for Medical Shop

- **Case Study:**

1. Angular JS
2. VueJS
3. Django
4. Flask
5. Wordpress

- **Blog:**

1. Recent Web Development Trends
2. Databases for Web Developers
3. Web Services
4. Web Security
5. Web Evolution

- **Surveys:**

1. Comparison of Web Services
2. Frameworks for Web Development
3. Scripting languages for Web Designing
4. Web Server Vs Application Server
5. [Current Technologies for Web Development](#)

Assessment Scheme:

- ***Suggest an Assessment scheme that is best suited for the course. Ensure 360 degree assessment and check if it covers all aspects of Blooms Taxonomy.***
- 1. Home Assignment : Design,Case study,Survey and Blog/Seminar/Group Discussion 20Marks
- 2. CVV 20Marks
- 3. LAB-Course Assignment 10Marks
- 4. Project Evaluation 20Marks
- 5. ESE MCQ 30Marks

Books

1. *Thomas A. Powell; “Complete reference HTML”; 4th edition, Tata McGraw-Hill Publications*
2. *Black book; “Web Technologies:HTML,JS,PHP,Java,JSP,ASP.NET,XML and AJAX” ; Dreamtech Press, 2016.*
3. *Dave Mercer, Allan Ken; “Beginning PHP 5”; Dreamtech Publications.*
4. *Martin, M.G., “Programming for Beginners: 6 Books in 1 – Swift+PHP+Java+Javascript+Html+CSS: Basic Fundamental Guide for Beginners”,*

Scheme of Marking for continuous assessment of Lab assignment

Sr. No.	Parameter	% of Marks
1	Regularity and Interaction (2 Marks)	20
2	Method based evaluation -Timely submission of assignment (2 Marks) -Programming/ coding style and guidelines used (2 Marks)	40
3	Practical Assignments -Originality of assignment (Innovativeness) (2 Marks) -Understanding the theme and use of assignment performed (2 Marks)	40

Introduction

Web Technology

- The methods by which computers communicate with each other through the use of markup languages and multimedia packages is known as **web technology**.

The Internet

- INTERconnected NETwork
- The Internet is a giant network of networks
- A network may include PCs, and other devices like servers or printers
- A network is connected through a communication channel
- The first long distance communication took place in 1965 between a computer in MIT and California.
- In 1969, four computers clients were connected together via ARPAnet.

How do we use the Internet?

- Email
 - WWW, hypertext, browsers
- RSS
- FTP, P2P file distribution
- Mobile Internet
- IM, IRC, Skype
 - Blogging, microblogging
- Gaming
- Learning
 - Learning
 - Video Conferencing
 - Remote Backup
 - Streaming video and audio
 - Collaboration-Participation (Wiki)
 - Collaborative tagging
 - Software over the web
 - Rich User Experiences
 - Social networks
 - Business and finance

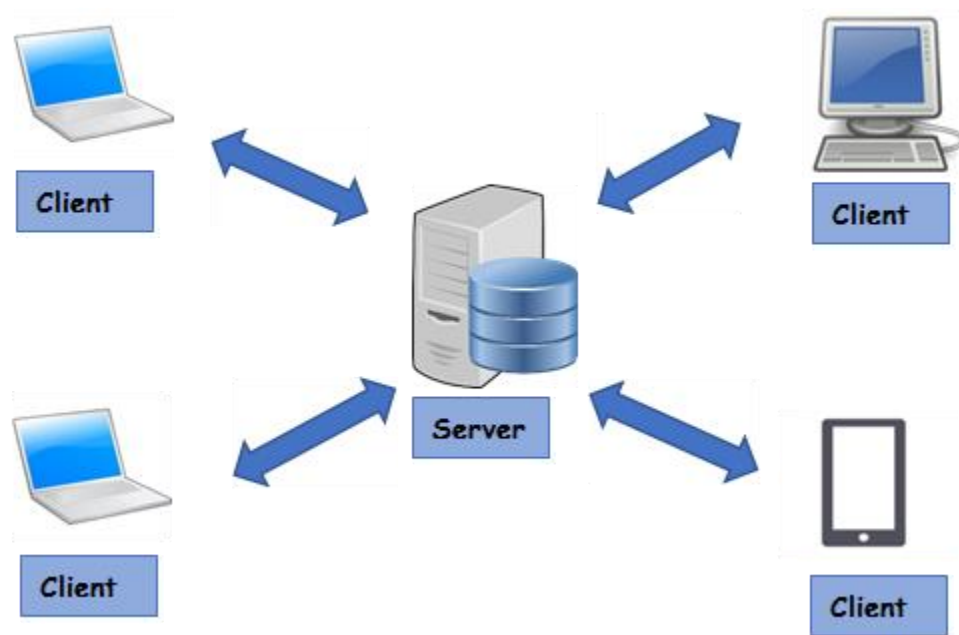
WWW

- World Wide Web (www.)
- The World Wide Web (WWW or Web) is often confused with the Internet
- It is the service used over the network.'It is the way of accessing information over the internet
- It uses HTTP protocol
- It uses different browsers to access web pages.
- The Web didn't exist until the 1980s
- In 1989 Tim Berners-Lee created a set of technologies that allowed information on the Internet to be linked together through the use of links, or connections in documents -
- The language used to write these documents with links is HTML

The Web

- The Web was mostly text based until Marc Andreessen created the Mosaic browser in 1992
- Accredited for popularizing the WWW
- People started thinking about adding videos, sound, and graphics on the Web.
- Now many people think of the Web as the graphical or illustrated part of the Internet

Client-server architecture



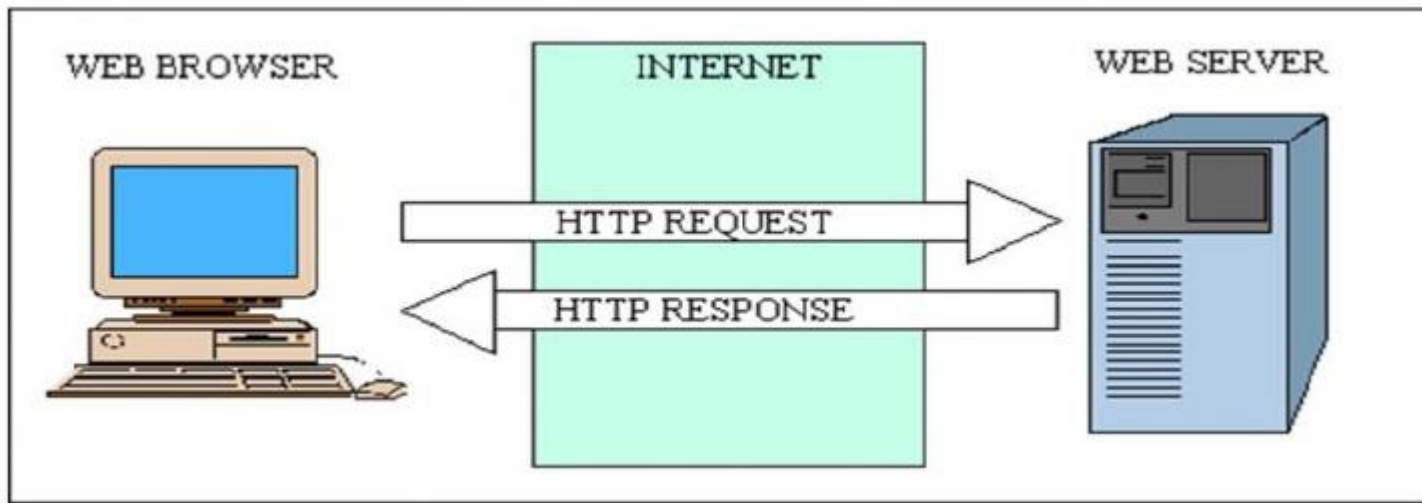
- **How the browser interacts with the servers ?**

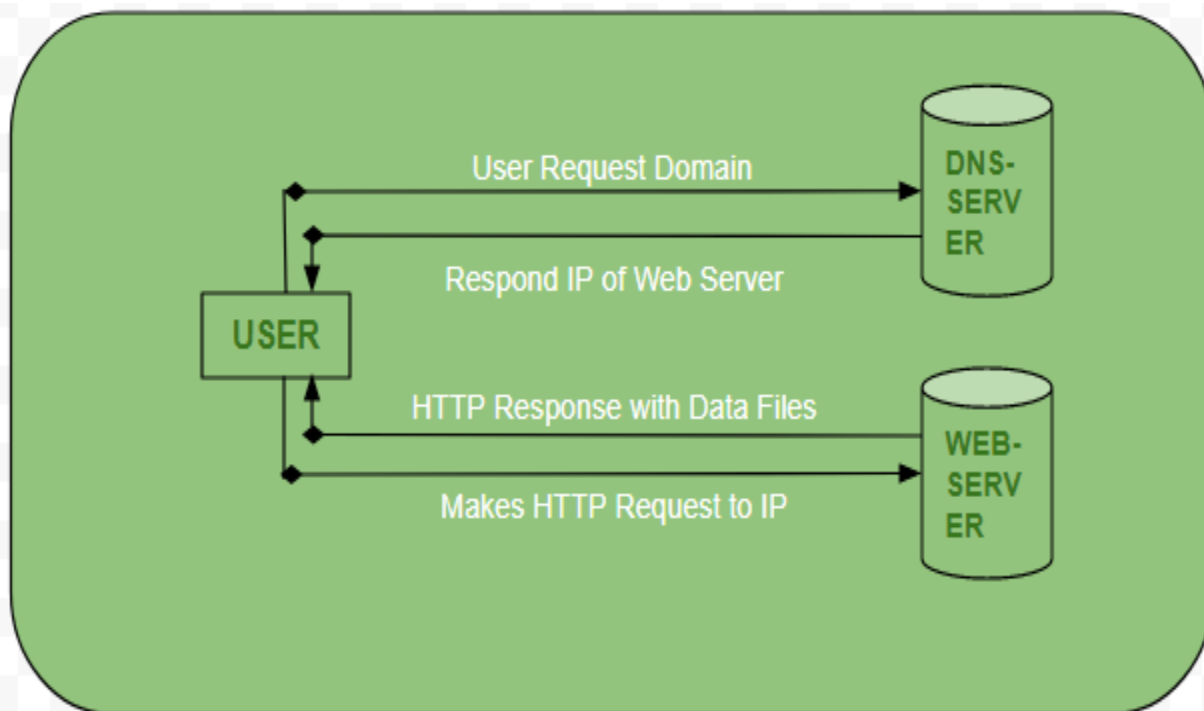
There are few steps to follow to interact with the servers as a client.

- User enters the **URL**(Uniform Resource Locator) of the website or file. The Browser then requests the **DNS**(DOMAIN NAME SYSTEM) Server.
- **DNS Server** lookup for the address of the **WEB Server**.
- **DNS Server** responds with the **IP address** of the **WEB Server**.
- Browser sends over an **HTTP/HTTPS** request to **WEB Server's IP** (provided by **DNS server**).
- Server sends over the necessary files of the website.
- Browser then renders the files and the website is displayed. This rendering is done with the help of **DOM** (Document Object Model) interpreter, **CSS** interpreter and **JS Engine** collectively known as the **JIT** or (Just in Time) Compilers.

Client-server architecture

- Example of Weather forecasting
- Client
- Server
- HTTP protocol-stateless request/reply protocol
- Errors-





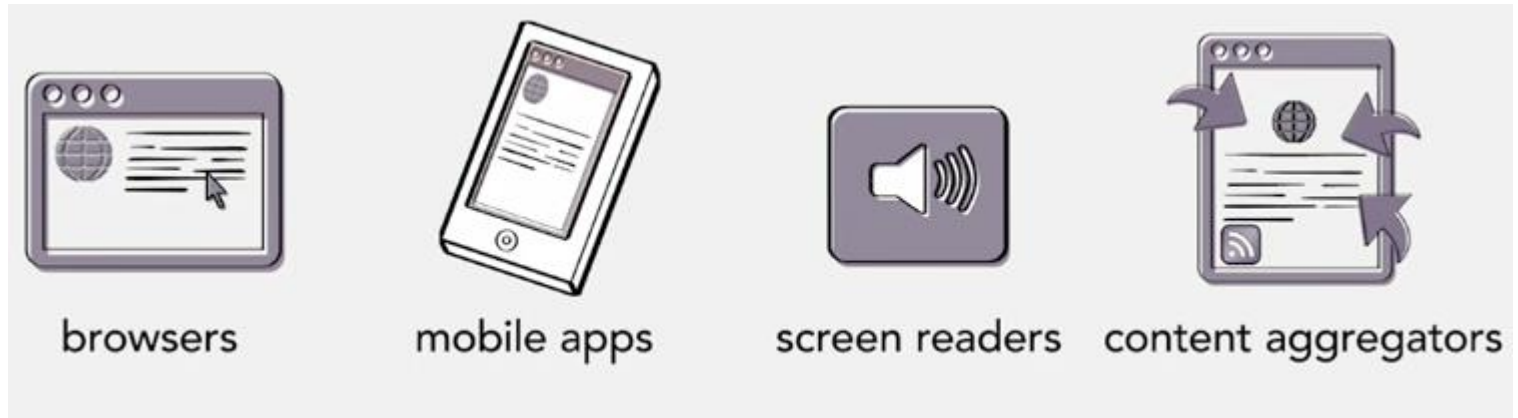
- **Advantages of Client-Server model:**

- Centralized system with all data in a single place.
- Cost efficient requires less maintenance cost and Data recovery is possible.
- The capacity of the Client and Servers can be changed separately.

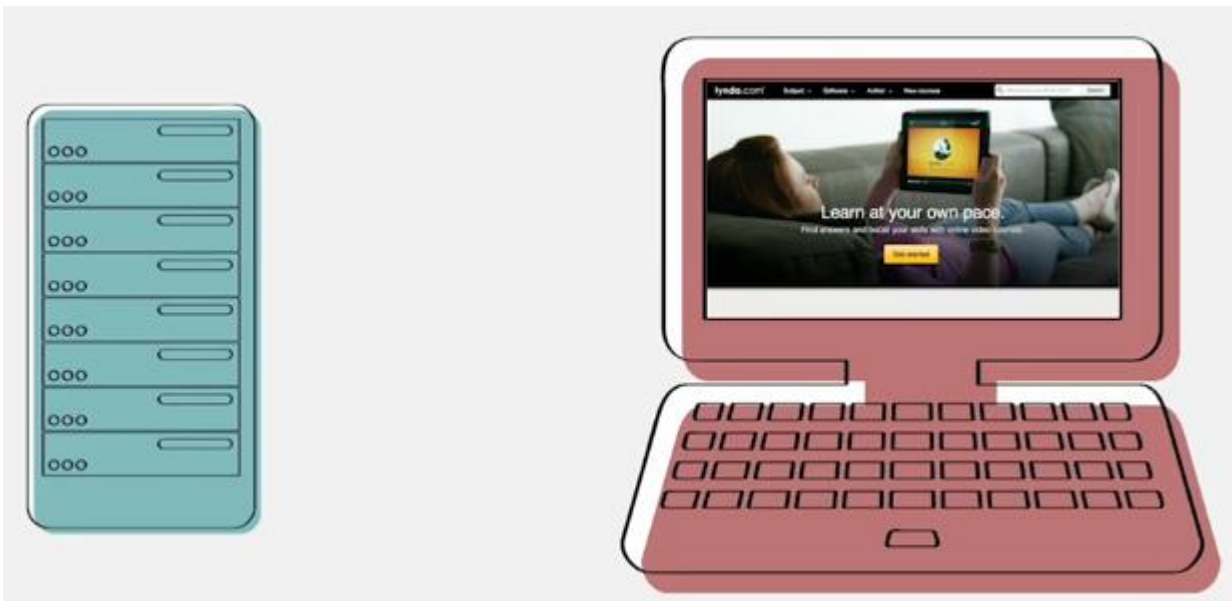
- **Disadvantages of Client-Server model:**

- Clients are prone to viruses, Trojans and worms if present in the Server or uploaded into the Server.
- Server are prone to Denial of Service (DOS) attacks.
- Data packets may be spoofed or modified during transmission.
- Phishing or capturing login credentials or other useful information of the user are common and MITM(Man in the Middle) attacks are common.

- CLIENT



- SERVER



Browsers

- An application that provides a way to look at and interact with the information on the World Wide Web
- It retrieves, presents, and traverses information resources
 - These include web pages, images, video, and other multimedia content



Design issues

- Browser & OS
- Bandwidth
- Display resolution
- Look and feel
- Page layout and linking
- User centric design
- **Planning a website:**
- Planning plays an important role in the successful completion of any work. Creation of website takes time and resources, so plan must proceed sequentially and properly. This plan contain the following things
 - Defining the purpose of website
 - Knowing the audience of website
 - Organizing contents of website
 - Publishing of website-***Domain registration and hosting, Upload website on server Viewing website***

- Good design involves:
 - Using understandable language on the web pages.
 - Making the site interesting
 - Making the site easy to use.
 - using a uniform look and feel

How to Publish a Website

1. Choose how you'll publish your website
2. Pick your domain name
3. Choose a web hosting provider
4. Create content for your website
5. Design your website
6. Conduct a QA audit
7. Publish your website
8. Prepare for the future



Src: <https://mailchimp.com/resources/how-to-publish-a-website/>

WWW

- The Web, or World Wide Web, is basically a system of Internet servers that support specially formatted documents. The documents are formatted in a markup language called HTML (*Hyper Text Markup Language*) that supports links to other documents, as well as graphics, audio, animations and video files.
- This means you can jump from one document to another simply by clicking on hot spots. Not all Internet servers are part of the World Wide Web. There are several applications called Web browsers that make it easy to access the World Wide Web; Two of the most popular being Firefox and Microsoft's Internet Explorer.

W3C



- W3C (World Wide Web consortium) is an international community where Member organizations, a full-time staff, and the public work together to develop Web standards.
- led by Tim Berners-Lee, inventor of the World Wide Web and Director and Dr. Jeffrey Jaffe, W3C CEO.
- W3C decided to develop non proprietary, interoperable technologies for the world wide web.
- The primary goal is to make the web universally accessible – regardless of ability, language or culture.
 - Web for all- enables human communication, commerce, and opportunities to share knowledge
 - Web on Everything- Mobile phones, smart phones, personal digital assistants, interactive television systems, voice response systems, kiosks
- W3C's vision for the Web involves participation, sharing knowledge, and thereby building trust on a global scale.

➤ For example

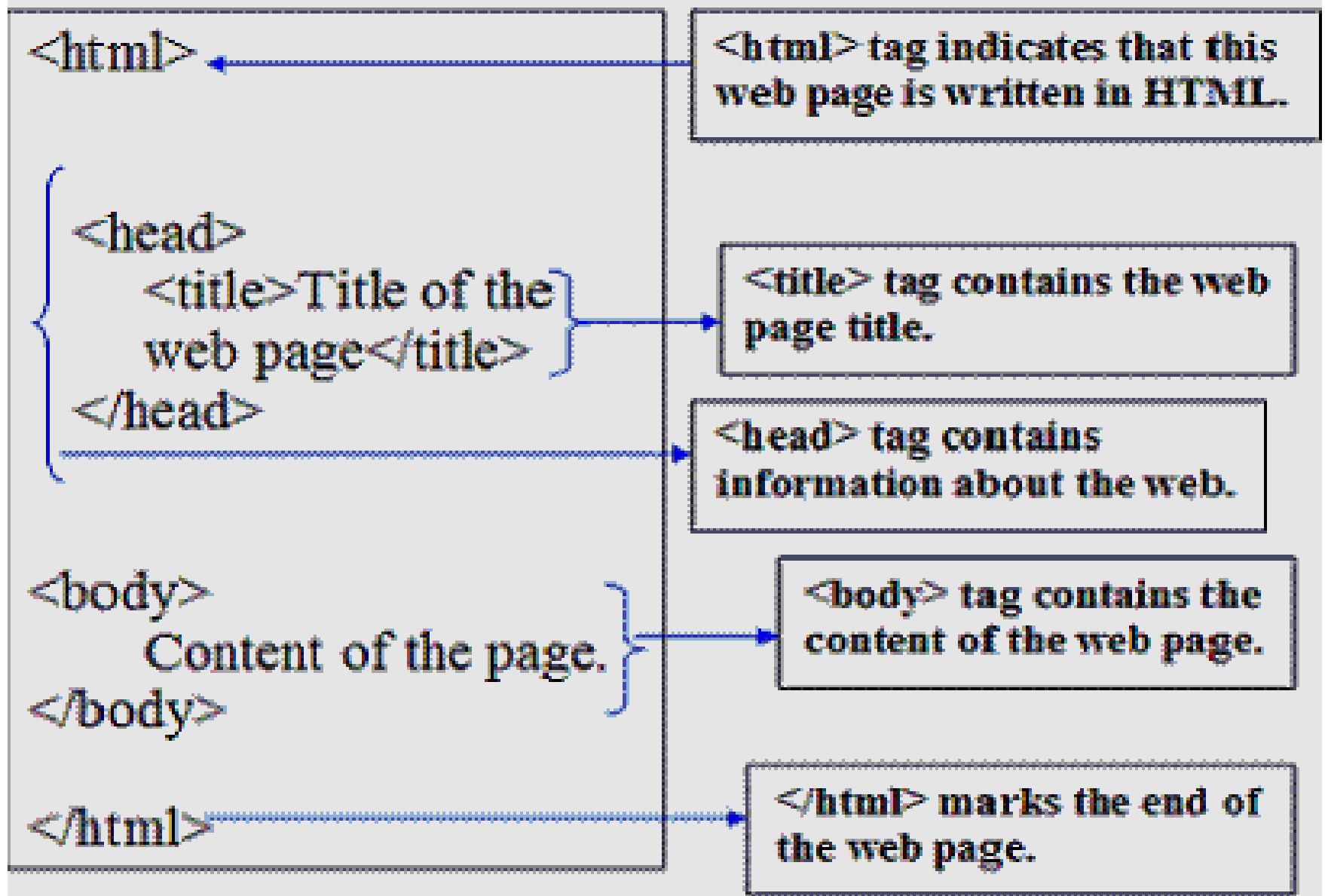
- If a writer wants to publish a business website promoting her professional writing services, she must acquire web publishing software in order for her to insert text, images and videos into an easily readable format suitable for the Internet. Then, she must locate a free or paid web server to deliver her content onto the Internet. Web servers are usually extremely high-powered computers that "store" websites and release the website content every time a user types in or clicks on the web address of the website. The writer must also have an Internet connection to upload, maintain and make changes to her website.

- **What is the difference between programming language and markup language?**
- Ans:- A programming language is an artificial language designed to communicate instruction to a machine, particularly a computer, whereas markup language is a language that is used to format the text inside the HTML document, so that the look and feel of a web page will increase. Programming languages can be used to create programs that control the behavior of a machine and/or to express algorithm precisely. The term Markup means the only particular text will change that is specified within the opening and closing tag. For instance, the `` tag instructs a browser to display the text that comes after it in bold text. To end the bold text, the `` tag is inserted.

What is HTML?

- HTML is a markup language for describing web documents (web pages).
- HTML stands for Hyper Text Markup Language.
- HTML was created by Berners-Lee in late 1991.
- Currently we are having HTML-5 published in 2012.
- A markup language is a set of markup tags.
- HTML documents are described by HTML tags.
- Each HTML tag describes different document content.
- Markup refers to the sequence of characters or other symbols that you insert at certain places in a text or word processing file to indicate how the file should look when it is printed or displayed or to describe the document's logical structure. The markup indicators are often called "tags."

HTML WEB PAGE STRUCTURE



HTML Element Syntax

- An HTML element starts with a **start tag** <> **opening tag**
- An HTML element ends with an **end tag** </> **closing tag**
- The **element content** is everything between the start and the end tag
- Some HTML elements have **empty content**
- Empty elements are **closed in the start tag**
- Most HTML elements can have **attributes**

Nested HTML Elements

Most HTML elements can be nested (can contain other HTML elements).

HTML documents consist of nested HTML elements.

HTML Document Example

```
<!DOCTYPE html>
<html>

  <body>
    <p>This is my first paragraph.</p>
  </body>

</html>
```

A small HTML document:

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

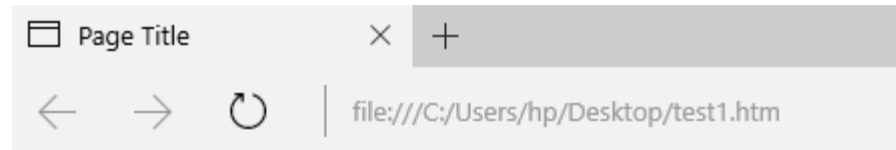
<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

18,48,49,24,21,65,61,14,17,25,42,51,57,55,60,52,54,58,15,56,46,38,36,47
,1,32,3,62,71,52,22,30,13,51,3167,63,30,26,70,35,41

How to create and run html file

1. Write some html code in notepad/notepad++
2. Save the file with extension .html or .htm
3. Open the file in browser



My First Heading

My first paragraph.

Example Explained

- The **DOCTYPE** declaration defines the document type to be HTML
- The text between **<html>** and **</html>** describes an HTML document
- The text between **<head>** and **</head>** provides information about the document
- The text between **<title>** and **</title>** provides a title for the document
- The text between **<body>** and **</body>** describes the visible page content
- The text between **<h1>** and **</h1>** describes a heading
- The text between **<p>** and **</p>** describes a paragraph
- Using this description, a web browser can display a document with a heading and a paragraph.

The HTML <head> Element

- The HTML <head> element contains **meta data**. Meta data are not displayed.
- The HTML <head> element is placed between the <html> tag and the <body> tag:
- The title will not be displayed in the document, but might be displayed in the browser tab.

NOTE: Meta data means data **about** data. HTML meta data is data **about** the HTML document.

HTML TAGS

- An HTML element is defined by a starting tag.
- So here `<p>....</p>` is an HTML element
- There are some HTML elements which don't need to be closed, such as `<img.../>`, `<hr />` and `
` elements. These are known as void elements.

HTML – ATTRIBUTES

- An attribute is used to define the characteristics of an HTML element and is placed inside the element's opening tag. All attributes are made up of two parts: a name and a value:
 - `<p align="left">This is left aligned</p>`
 - `<p align="center">This is center aligned</p>`
 - `<p align="right">This is right aligned</p>`
- Core Attributes
 - `<h3 title="Hello HTML!">Titled Heading Tag Example</h3>`
- The style Attribute
 - `<p style="font-family:arial; color:#FF0000;">Some text...</p>`
- The dir Attribute
 - `<html dir="rtl">`

HTML – FORMATTING

Text formatting tags

HTML Code	Sample
bold	bold
<U>underline</U>	<u>underline</u>
<I>italic</I>	<i>italic</i>

Alignment tags

HTML Code	Sample
<P ALIGN=Left> your text	your text
<P ALIGN=Center> your text	your text
<P ALIGN=Right> your text	your text

Break tags

HTML Code	Description
<P>	Paragraph break

	Forced line break

- *Character Tags*

- A tag that you apply to an individual character is referred to as a character tag. There are two types of character tags, physical and logical.
- **Physical Tags**
- Physical tags are used to indicate exactly how specific characters are to be formatted.

The syntax or format for using a PHYSICAL TAG is as follows:

<Tag Name> Characters to be formatted. </Tag Name>

-

	Indicates that the text should be bold.	
<I>	<i>Indicates that the text should be italic.</i>	</I>
<TT>	Indicates that the text should be used with a font such as Courier that allots the same width to each character.	</TT>
<BIG>	Indicates that the text should be displayed in a big font. Available in HTML 3.0 or higher.	</BIG>
<SMALL>	Indicates that the text should be displayed in a small font. Available in HTML 3.0 or higher.	</SMALL>
_{	Indicates that the text should be displayed as a _{subscript} , in a smaller font if possible. Available in HTML 3.0 or higher.	}
^{	Indicates that the text should be displayed as a ^{superscript} , in a smaller font if possible. Available in HTML 3.0 or higher.	}
<U>	<u>Indicates that the text should be displayed underlined. Not all browsers support this tag.</u>	</U>
	Indicates that the text should be bold.	

- **Logical Tags**

- Logical tags are used to indicate to the visually impaired that there is some emphasizes on the text. Each browser has its own technique as to how to indicate to its viewer that the text between the tags are different. The syntax or format for using a LOGICAL TAG is as follows:

<Tag Name> Character/s to be formatted. </Tag Name>

PHRASE TAGS

- The phrase tags have been designed for specific purposes, though they are displayed in a similar way as other basic tags like ``, `<i>`, `<pre>`, and `<tt>`,
- **Emphasized Text** Anything that appears within `...` element is displayed as emphasized text.
 - `<p>The following word uses a emphasized typeface.</p>`
- **Marked Text** : Anything that appears with-in `<mark>...</mark>` element, is displayed as marked with yellow ink
 - `<p>The following word has been marked with yellow</p>`
- **Strong Text** Anything that appears within `...` element is displayed as important text.
 - `<p>The following word uses a strong typeface.</p>`

- Text Abbreviation You can abbreviate a text by putting it inside opening `<abbr>` and closing `</abbr>` tags. If present, the title attribute must contain this full description and nothing else.
 - `<p>My best fiend's name is <abbr title="Abhishek">Abhy</abbr>.</p>`
- Acronym Element The `<acronym>` element allows you to indicate that the text between `<acronym>` and `</acronym>` tags is an acronym.
 - `<p>This chapter covers marking up text in <acronym>XHTML</acronym>.</p>`
- Text Direction The `<bdo>...</bdo>` element stands for Bi-Directional Override and it is used to override the current text direction.
 - `<p>This text will go left to right.</p>`
 - `<p><bdo dir="rtl">This text will go right to left.</bdo></p>`
- Special Terms The `<dfn>...</dfn>` element (or HTML Definition Element) allows you to specify that you are introducing a special term. It's usage is similar to italic words in the midst of a paragraph.
 - `<p>The following word is a <dfn>special</dfn> term.</p>`

- Quoting Text When you want to quote a passage from another source, you should put it in between `<blockquote>...</blockquote>` tags.
 - `<blockquote>XHTML 1.0 is the W3C's first Recommendation for XHTML, following on from earlier work on HTML 4.01, HTML 4.0, HTML 3.2 and HTML 2.0.</blockquote>`
- Short Quotations The `<q>...</q>` element is used when you want to add a double quote within a sentence.
 - `<p>Amit is in Spain, <q>I think I am wrong</q>.</p>`
- Text Citations If you are quoting a text, you can indicate the source placing it between an opening `<cite>` tag and closing `</cite>` tag
 - `<p>This HTML tutorial is derived from <cite>W3 Standard for HTML</cite>.</p>`
- Computer Code Any programming code to appear on a Web page should be placed inside `<code>...</code>` tags. Usually the content of the `<code>` element is presented in a monospaced font, just like the code in most programming books.
 - `<p>Regular text. <code>This is code.</code> Regular text.</p>`

- Keyboard Text When you are talking about computers, if you want to tell a reader to enter some text, you can use the `<kbd>...</kbd>` element to indicate what should be typed in, as in this example.
 - `<p>Regular text. <kbd>This is inside kbd element</kbd> Regular text.</p>`
- Address Text The `<address>...</address>` element is used to contain any address.
 - `<address>388A, Road No 22, Jubilee Hills - Hyderabad</address>`

HTML – META TAGS

- HTML lets you specify metadata - additional important information about a document in a variety of ways. The META elements can be used to include name/value pairs describing properties of the HTML document, such as author, expiry date, a list of keywords, document author etc.
- The <meta> tag is used to provide such additional information. This tag is an empty element
- Specify Character Set You can use <meta> tag to specify character set used within the webpage
- <meta charset=UTF-8" />

More Meta Elements

In the chapter about HTML styles you discover more meta elements:

- The HTML **<style>** element is used to define internal CSS style sheets.
- The HTML **<link>** element is used to define external CSS style sheets.

HTML Comments

- Comments are not displayed by the browser, but they can help document your HTML.
- With comments you can place notifications and reminders in your HTML.
- Comment tags `<!--` and `-->` are used to insert comments in HTML.
 - `<!-- This is valid comment -->`
 - `< !-- This is not a valid comment -->`
- Because there is a space between the left angle bracket and the exclamation mark
- Multiline Comments
 - `<!--`
 - This is a multiline comment and it can
 - span through as many as lines you like.
 - `-->`

HTML Line Breaks

- The HTML **
** element defines a **line break**.
- Use **
** if you want a line break (a new line) without starting a new paragraph:

Example:

```
<p>This is<br>a para<br>graph with line breaks</p>
```

O/P:

This is
a para
graph with line breaks

- **NOTE:** The **
** element is an empty HTML element. It has no end tag.

The HTML <pre> Element

<pre>

Twinkle, twinkle, little star,

How I wonder what you are!

Up above the world so high,

Like a diamond in the sky. 😊

</pre>

O/P:

Twinkle, twinkle, little star,

How I wonder what you are!

Up above the world so high,

Like a diamond in the sky. 😊

The pre tag preserves
both spaces and line
breaks

HTML Headings

- Headings Are Important
- Use HTML headings for headings only. Don't use headings to make text **BIG** or **bold**.
- Search engines use your headings to index the structure and content of your web pages.
- `<h1>` defines the most important heading. `<h6>` defines the least important heading.

Example:

<h1>This is a heading</h1>

<h2>This is a heading</h2>

<h3>This is a heading</h3>

<h4>This is a heading</h4>

<h5>This is a heading</h5>

<h6>This is a heading</h6>

This is a heading 1

This is a heading 2

This is a heading 3

This is a heading 4

This is a heading 5

This is a heading 6

HTML Horizontal Rules

- The **<hr>** tag creates a horizontal line in an HTML page.
- The hr element can be used to separate content:

Example:

```
<p>This is a paragraph.</p>
```

```
<hr>
```

```
<p>This is a paragraph.</p>
```

```
<hr>
```

```
<p>This is a paragraph.</p>
```

O/P:

This is a paragraph.

This is a paragraph.

This is a paragraph.

Changing background color of page

The background-color property defines the background color for an element:

```
<!DOCTYPE html>
```

```
<html>
```

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

```
<h2 style="background-color:red;">This is a heading</h2>
```

```
<p style="background-color:green;">This is a paragraph.</p>
```

```
</body>
```

```
</html>
```

008822947

changing background color of page, text color

Background Color:

```
<body bgcolor="#E6E6FA">
```

NOTE: The <body> bgcolor attribute is not supported in HTML5. Use CSS instead.

Text Color:

```
<font size="3" color="red">This is some text!</font>
```

```
<font size="2" color="blue">This is some text!</font>
```

```
<font face="verdana" color="green">This is some text!</font>
```

Not Supported in HTML5.

HTML Tables

Student information:

Number	First Name	Last Name	Percentage
1.	Aman	Patel	87%
2.	Mala	Dhole	55%
3.	Sarita	Patil	90%

- For Table mainly used tags are `<table>`, `<tr>` and `<td>`
- Tables are defined with the **`<table>`** tag.
- `<tr>` - For table row (tables are divided into table rows)
- `<td>` - For table column (table rows are divided into table data)
- `<th>` - For the table header


```
<!DOCTYPE html>
<html>
<head>
<title>HTML Table Header</title>
</head>
<body>
<caption>table</caption>
<table border="1">
<tr>
<th>Name</th>
<th>Salary</th>
</tr>
<tr>
<td>Ramesh Raman</td>
<td>5000</td>
</tr>
<tr>
<td>Shabbir Hussein</td>
<td>7000</td>
</tr>
</table>
</body>
</html>
```

table

Name	Salary
Ramesh Raman	5000
Shabbir Hussein	7000

Cellpadding and Cellspacing Attributes:

- The cellspacing attribute defines the width of the border
- cellpadding represents the distance between cell borders and the content within a cell

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Table
Cellpadding</title>
</head>
<body>
<table border="1"
cellpadding="5"
cellspacing="5">
<tr>
<th>Name</th>
<th>Salary</th>
</tr>
<tr>
<td>Ramesh Raman</td>
<td>5000</td>
</tr>
<tr>
<td>Shabbir Hussein</td>
<td>7000</td>
</tr>
</table>
</body>
</html>
```

Name	Salary
Ramesh Raman	5000
Shabbir Hussein	7000

Colspan and Rowspan Attributes

- You will use colspan attribute if you want to merge two or more columns into a single column. Similar way you will use rowspan if you want to merge two or more rows

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Table Colspan/Rowspan</title>
</head>
<body>
<table border="1">
<tr>
<th>Column 1</th>
<th>Column 2</th>
<th>Column 3</th>
</tr>
```

```
<tr>
<td rowspan="2">Row 1 Cell 1</td>
<td>Row 1 Cell 2</td>
<td>Row 1 Cell 3</td>
</tr>
<tr><td>Row 2 Cell 2</td>
<td>Row 2 Cell 3</td>
</tr>
<tr><td colspan="3">Row 3 Cell 1</td>
</tr>
</table>
</body>
</html>
```

Tables Backgrounds

- You can set table background using one of the following two ways:
 - bgcolor attribute - You can set background color for whole table or just for one cell.
 - background attribute - You can set background image for whole table or just for one cell.

```
<table border="1" bordercolor="red" bgcolor="green">
```

```
<tr>
```

```
<th>Column 1</th>
```

```
<th>Column 2</th>
```

```
<th>Column 3</th>
```

```
</tr>
```

```
<tr><td rowspan="2">Row 1 Cell 1</td><td>Row 1 Cell 2</td><td>Row 1 Cell 3</td></tr>
```

```
<tr><td>Row 2 Cell 2</td><td>Row 2 Cell 3</td></tr>
```

```
<tr><td colspan="3">Row 3 Cell 1</td></tr>
```

```
</table>
```

Column 1	Column 2	Column 3
Row 1 Cell 1	Row 1 Cell 2	Row 1 Cell 3
	Row 2 Cell 2	Row 2 Cell 3
Row 3 Cell 1		

```
<table border="1" bordercolor="green"  
background="/images/test.png">
```

```
<tr>
```

```
<th>Column 1</th>
```

```
<th>Column 2</th>
```

```
<th>Column 3</th>
```

```
</tr>
```

```
<tr><td rowspan="2">Row 1 Cell 1</td><td>Row 1 Cell 2</td><td>Row  
1 Cell 3</td></tr>
```

```
<tr><td>Row 2 Cell 2</td><td>Row 2 Cell 3</td></tr>
```

```
<tr><td colspan="3">Row 3 Cell 1</td></tr>
```

```
</table>
```

Table Height and Width

- You can set a table width and height using width and height attributes. You can specify table width or height in terms of pixels or in terms of percentage of available screen area.

```
<table border="1" width="400" height="150">  
<tr>  
<td>Row 1, Column 1</td>  
<td>Row 1, Column 2</td>  
</tr>  
<tr>  
<td>Row 2, Column 1</td>  
<td>Row 2, Column 2</td>  
</tr>  
</table>
```

Table Caption

- The caption tag will serve as a title or explanation for the table and it shows up at the top of the table.

```
<table border="1" width="100%">  
<caption>This is the caption</caption>  
<tr>  
<td>row 1, column 1</td><td>row 1, column 2</td>  
</tr>  
<tr>  
<td>row 2, column 1</td><td>row 2, column 2</td>  
</tr>  
</table>
```

Table Header, Body, and Footer

- Tables can be divided into three portions: a header, a body, and a foot.
- The three elements for separating the head, body, and foot of a table are:
- `<thead>` - to create a separate table header.
- `<tbody>` - to indicate the main body of the table.
- `<tfoot>` - to create a separate table footer.
- A table may contain several `<tbody>` elements to indicate different pages or groups of data. But it is notable that `<thead>` and `<tfoot>` tags should appear before `<tbody>`


```
<table border="1" width="100%">
<thead>
<tr>
<td colspan="4">This is the head of the
table</td>
</tr>
</thead>
<tfoot>
<tr>
<td colspan="4">This is the foot of the
table</td>
</tr>
</tfoot>
<tbody>
<tr>
<td>Cell 1</td>
<td>Cell 2</td>
<td>Cell 3</td>
<td>Cell 4</td>
</tr>
</tbody>
</table>
```

This is the head of the table			
Cell 1	Cell 2	Cell 3	Cell 4
This is the foot of the table			

Nested Tables

- You can use one table inside another table. Not only tables you can use almost all the tags inside table data tag <td>.

```
<table border="1" width="100%">
<tr>
<td>
  <table border="1" width="100%">
    <tr>
      <th>Name</th>
      <th>Salary</th>
    </tr>
    <tr>
      <td>Ramesh Raman</td>
      <td>5000</td>
    </tr>
    <tr>
      <td>Shabbir Hussein</td>
      <td>7000</td>
    </tr>
  </table>
</td>
</tr>
</table>
```

Name	Salary
Ramesh Raman	5000
Shabbir Hussein	7000

Table Cells that Span Many Columns

To make a cell span more than one column, use the **colspan** attribute:

```
<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>
```

Name	Telephone	
Bill Gates	555 77 854	555 77 855

Table Cells that Span Many Rows

To make a cell span more than one row, use the **rowspan** attribute:

```
<table border="1">
  <tr>
    <th>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>
```

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

Student information

<table style="width:100%">

<caption>Student Information</caption>

<tr>
 <th>Number</th>
 <th>First Name:</th>
 <th>Last Name:</th>
 <th> Percentage</th>
</tr>

<tr>
 <td>1.</td>
 <td>Aman</td>
 <td>Patel</td>
 <td> 87%</td>

</tr>

<tr>
 <td>2.</td>
 <td>Mala</td>
 <td>Dhole</td>
 <td> 55%</td>

</tr>

<tr>

<td>3.</td>

<td>Sarita</td>

<td>Patil</td>

<td> 90%</td>

</tr>

</table>

Student Information

Number First Name: Last Name: Percentage

1. Aman Patel 87%

2. Mala Dhole 55%

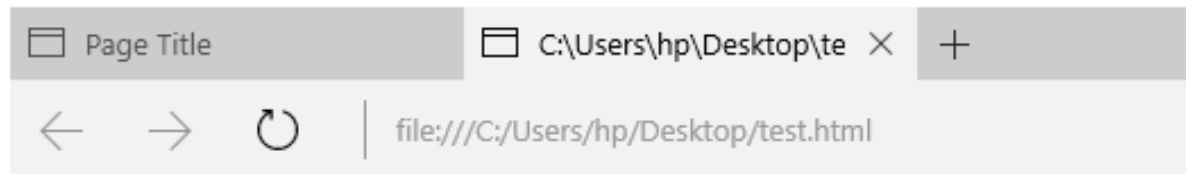
3. Sarita Patil 90%

- [<colgroup>](#) Specifies a group of one or more columns in a table for formatting
- [<col>](#) Specifies column properties for each column within a <colgroup> element

```
<head>
<style>
table, th, tr, td {
    border: 1px solid red;
}
</style>
</head>
<body>

<table>
  <colgroup>
    <col span="3" style="background-color:green">
    <col style="background-color:yellow">
  </colgroup>
  <tr>
    <th>Book</th>
    <th>Title</th>
    <th>Price</th>
  </tr>
  <tr>
    <td>HTML</td>
    <td>My first HTML</td>
    <td>253</td>
  </tr>

  <tr>
    <td>CSS</td>
    <td>My first CSS</td>
    <td>149</td>
  </tr>
</table>
```



Book	Title	Price
HTML	My first HTML	253
CSS	My first CSS	149

HTML Lists

- Unordered lists and ordered lists are commonly used in HTML

Unordered List	Ordered List
<ul style="list-style-type: none">•The first item•The second item•The third item•The fourth item	<ol style="list-style-type: none">1. The first item2. The second item3. The third item4. The fourth item

Unordered HTML Lists

- An unordered list starts with the **** tag.
- Each list item starts with the **** tag.
- The list items will be marked with bullets (small black circles):

Unordered List with Default Bullets

Example

```
<ul>  
  <li>Coffee</li>  
  <li>Tea</li>  
  <li>Milk</li>  
</ul>
```

O/P:

Unordered list

- Coffee
- Tea
- Milk

Unordered HTML Lists - The Style Attribute

Value	Description
disc	Sets the list item marker to a bullet (default)
circle	Sets the list item marker to a circle
square	Sets the list item marker to a square
none	The list items will not be marked
Value	Description

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Unordered List</title>
</head>
<body>
  <ul type="square">
    <li>Beetroot</li>
    <li>Ginger</li>
    <li>Potato</li>
    <li>Radish</li>
  </ul>
</body>
</html>
```

- Beetroot
- Ginger
- Potato
- Radish

Ordered HTML Lists

An ordered list starts with the `` tag. Each list item starts with the `` tag.

Example:

```
<ol>  
  <li>Coffee</li>  
  <li>Tea</li>  
  <li>Milk</li>  
</ol>
```

Ordered List

O/P:

Ordered list

1. Coffee

2. Tea

3. Milk

Ordered HTML Lists - The Type Attribute

A **type** attribute can be added to an **ordered list**, to define the type of the marker:

Type	Description
type="1"	The list items will be numbered with numbers (default)
type="A"	The list items will be numbered with uppercase letters
type="a"	The list items will be numbered with lowercase letters
type="I"	The list items will be numbered with uppercase roman numbers
type="i"	The list items will be numbered with lowercase roman numbers

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
  <ol type="1">
    <li>Beetroot</li>
    <li>Ginger</li>
    <li>Potato</li>
    <li>Radish</li>
  </ol>
</body>
</html>
```

1. Beetroot
2. Ginger
3. Potato
4. Radish

- The start Attribute You can use start attribute for tag to specify the starting point of numbering you need. Following are the possible options:
 - <ol type="1" start="4"> - Numerals starts with 4.
 - <ol type="I" start="4"> - Numerals starts with IV.
 - <ol type="i" start="4"> - Numerals starts with iv.
 - <ol type="a" start="4"> - Letters starts with d.
 - <ol type="A" start="4"> - Letters starts with D.


```
<!DOCTYPE html>
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
  <ol type="i" start="4">
    <li>Beetroot</li>
    <li>Ginger</li>
    <li>Potato</li>
    <li>Radish</li>
  </ol>
</body>
</html>
```

- iv. Beetroot
- v. Ginger
- vi. Potato
- vii. Radish

Nested HTML Lists

List can be nested (lists inside lists):

Example:

```
<ul>  
  <li>Coffee</li>  
  <li>Tea  
    <ul>  
      <li>Black tea</li>  
      <li>Green tea</li>  
    </ul>  
  </li>  
  <li>Milk</li>  
</ul>
```

A Nested List:

O/P:

- Coffee
- Tea
 - Black tea
 - Green tea
- Milk

HTML Definition Lists

- HTML and XHTML supports a list style which is called definition lists where entries are listed like in a dictionary or encyclopedia. The definition list is the ideal way to present a glossary, list of terms, or other name/value list.
- Definition List makes use of following three tags.
 - `<dl>` - Defines the start of the list
 - `<dt>` - A term
 - `<dd>` - Term definition
 - `</dl>` - Defines the end of the list

```
<html>
<head>
<title>HTML Definition List</title>
</head>
<body>
<dl>
<dt><b>HTML</b></dt>
<dd>This stands for Hyper Text Markup Language</dd>
<dt><b>HTTP</b></dt>
<dd>This stands for Hyper Text Transfer Protocol</dd>
</dl>
</body>
</html>
```

HTML

This stands for Hyper Text Markup Language

HTTP

This stands for Hyper Text Transfer Protocol

Summary

- Use the HTML **** element to define an unordered list
- Use the HTML **type** attribute to define the bullet style
- Use the HTML **** element to define an ordered list
- Use the HTML **type** attribute to define the numbering type
- Use the HTML **** element to define a list item
- Use the HTML **<dl>** element to define a definition list
- Use the HTML **<dt>** element to define the definition term
- Use the HTML **<dd>** element to define the definition data

HTML Link

- A webpage can contain various links that take you directly to other pages and even specific parts of a given page. These links are known as hyperlinks
- Navigate between Web sites by clicking on words, phrases, and images.
- A link is specified using HTML tag <a>. This tag is called anchor tag .
- Anything between the opening <a> tag and the closing tag becomes part of the link and a user can click that part to reach to the linked document HTML link is a hyperlink.
- Syntax:
`link text`
`Welcome to gmail`

The **href** attribute specifies the destination address

Local Links

- The example above used an absolute URL (A full web address).
- A local link (link to the same web site) is specified with a relative URL (without <http://www....>).

```
<a href="list.html">HTML Images</a>
```

HTML Links - The target Attribute

The target Attribute: This attribute is used to specify the location where linked document is opened

Option	Description
<code>_blank</code>	Opens the linked document in a new window or tab.
<code>_self</code>	Opens the linked document in the same frame.
<code>_parent</code>	Opens the linked document in the parent frame.
<code>_top</code>	Opens the linked document in the full body of the window.
<code>targetframe</code>	Opens the linked document in a named <i>targetframe</i> .

```
<a href="http://www.vit.edu/" target="_blank">VIT College</a>
```

```
<a href="http://www.vit.edu/" target="_self">VIT College</a>
```



```
<!DOCTYPE html>
<html>
<head>
<title>Hyperlink Example</title>
</head>
<body>
<p>Click any of the following links</p>
<p>Click any of the following links</p>
<a href="/2018-19/web technology/list.html" target="_blank">Opens in New</a> <br>
<a href="/2018-19/web technology/list.html" target="_self">Opens in Self</a> <br>
<a href="/2018-19/web technology/list.html" target="_parent">Opens in Parent</a> <br>
<a href="/2018-19/web technology/list.html" target="_top">Opens in Body</a>
</body>
</html>
```

Click any of the following links

[Opens in New](#)

[Opens in Self](#)

[Opens in Parent](#)

[Opens in Body](#)

- Use of Base Path When you link HTML documents related to the same website, it is not required to give a complete URL for every link. You can get rid of it if you use <base> tag in your HTML document header.
- This tag is used to give a base path for all the links. So your browser will concatenate given relative path to this base path and will make a complete URL.

```
<!DOCTYPE html>
<html>
<head>
<title>Hyperlink Example</title>
<base href="http://www.tutorialspoint.com/">
</head>
<body>
<p>Click following link</p>
<a href="/html/index.htm" target="_blank">HTML Tutorial</a>
</body>
</html>
```

[HTML Tutorial](#)

Setting Link Colors:

- You can set colors of your links, active links and visited links using link, alink and vlink attributes of <body> tag.
 - By default, a link will appear like this (in all browsers):
 - An unvisited link is underlined and blue
 - A visited link is underlined and purple
 - An active link is underlined and red
- Example

```
<!DOCTYPE html>
<html>
<head>
<title>Hyperlink Example</title>
<base href="http://www.tutorialspoint.com/">
</head>
<body alink="yellow" link="green" vlink="cyan">
<p>Click following link</p>
<a href="/html/index.htm" target="_blank" >HTML Tutorial</a>
</body>
</html>
```

Click following link
[HTML Tutorial](#)

- Download Links :You can create text link to make your PDF, or DOC or ZIP files downloadable.
- Example:

```
!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Hyperlink Example</title>
```

```
</head>
```

```
<body>
```

```
<a href="http://www.tutorialspoint.com/page.pdf">Download PDF File</a>
```

```
</body>
```

```
</html>
```

[Download PDF File](http://www.tutorialspoint.com/page.pdf)

HTML – IMAGE LINKS

- The tag is empty, it contains attributes only, and does not have a closing tag.
- The src attribute specifies the URL (web address) of the image:

```

```

```

```

```

```

- Set Image Border By default, image will have a border around it, you can specify border thickness in terms of pixels using border attribute. A thickness of 0 means, no border around the picture.
 -
- Set Image Alignment By default, image will align at the left side of the page, but you can use align attribute to set it in the center or right.
 -

HTML Links - Image as Link

<p>The image is a link. You can click on it.</p>

alt Attribute : The alt attribute specifies an alternate text for an image, if the image cannot be displayed.

The image is a link. You can click on it.



HTML – EMAIL LINKS

- HTML <a> tag provides you option to specify an email address to send an email.
- While using <a> tag as an email tag, you will use mailto: email address along with href attribute.
- Following is the syntax of using mailto instead of using http.
- Send Email

[Send Email](mailto:abc@example.com)

HTML Links - Create a Bookmark

- HTML bookmarks are used to allow readers to jump to specific parts of a Web page.
- Useful if your webpage is very long.
- To make a bookmark, you must first create the bookmark, and then add a link to it.
- When the link is clicked, the page will scroll to the location with the bookmark.
- NOTE: only useful when no of links are more.

(Won't work in 2-3 links)

Example of Bookmark Link

```
<p><a href="#C2">Jump to Email  
Server2</a></p>
```

```
<h2 id="C1">Email Server1</h2>  
<p>This chapter explains ba bla bla</p>
```

```
<h2 id="C2">Email Server2</h2>  
<p>This chapter explains ba bla bla</p>
```

```
<h2 id="C3">Email Sever3</h2>  
<p>This chapter explains ba bla bla</p>
```

```
<h2>Email Sever4</h2>  
<p>This chapter explains ba bla bla</p>
```

```
<h2>Email Sever5</h2>  
<p>This chapter explains ba bla bla</p>
```

[Jump to Email Server2](#)

Email Server1

This chapter explains ba bla bla

Email Server2

This chapter explains ba bla bla

Email Sever3

This chapter explains ba bla bla

Email Sever4

This chapter explains ba bla bla

Email Sever5

This chapter explains ba bla bla

Email Sever6

This chapter explains ba bla bla

Summery

- Use the HTML **<a>** element to define a link
- Use the HTML **href** attribute to define the link address
- Use the HTML **target** attribute to define where to open the linked document
- Use the HTML **** element (inside **<a>**) to use an image as a link
- Use the HTML **id** attribute (**id="value"**) to define bookmarks in a page
- Use the HTML **href** attribute (**href="#value"**) to link to the bookmark
- Use the **alt** attribute to show text while failure of image display

HTML – FRAMES

- HTML frames are used to divide your browser window into multiple sections where each section can load a separate HTML document.
- A collection of frames in the browser window is known as a frameset.
- The window is divided into frames in a similar way the tables are organized: into rows and columns.
- Disadvantages of Frames:
 1. Some smaller devices cannot cope with frames often because their screen is not big.
 2. Page will be displayed differently on different computers due to different screen resolution.
 3. The browser's back button might not work as the user hopes.
 4. Few browsers that do not support frame technology.

Creating Frames

- Use <frameset> tag instead of <body> tag.
- It defines, how to divide the window into frames. The rows attribute of <frameset> tag defines horizontal frames and cols attribute defines vertical frames.
- Each frame is indicated by <frame> tag and it defines which HTML document shall open into the frame.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>HTML Frames</title>
```

```
</head>
```

```
<frameset cols="30%,40%,30%">
```

```
  <frame name="top" src="page1.html" />
```

```
  <frame name="main" src="page2.html" />
```

```
  <frame name="bottom" src="page3.html" />
```

```
  <noframes>
```

```
    <body>    Your browser does not support frames.  </body>
```

```
  </noframes>
```

```
</frameset>
```

```
</html>
```

<div>FILE NAME X</div> <div>< > ↺ file:///C:/Users/ADMIN/Desktop/</div> <div>☆ = ✕ 🔔 ...</div>		
WEB TECHNOLOGY	DBMS	C programming
This is WT Subject	This is DBMS subject	This is C programming subject

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Frames</title>
</head>
<frameset rows="30%,40%,30%">
  <frame name="top" src="page1.html" />
  <frame name="main" src="page2.html" />
  <frame name="bottom" src="page3.html" />
  <noframes>
    <body>    Your browser does not support frames.
  </body>
  </noframes>
</frameset>
</html>
```



WEB TECHNOLOGY

This is WT Subject

DBMS

This is DBMS subject

C programming

This is C programming subject



Nested frames

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Frames</title>
</head>
<frameset cols="20%,20%,20%,*">
  <frame name="f1" src="page1.html" />
  <frame name="f2" src="page2.html" />
  <frame name="f3" src="page3.html" />
</frameset rows="30%,40%,30%">
  <frame name="top" src="page1.html" />
  <frame name="main" src="page2.html" />
  <frame name="bottom" src="page3.html" />

  <noframes>
    <body>    Your browser does not support frames.  </body>
  </noframes>
</frameset>
</html>
```


HTML Frames

✕

+

←

→

↺

file:///G:/2017-18/IT/frame_nest.HTML

📖

☆

≡

🔍

🔔

⋮

WEB TECHNOLOGY This is WT Subject	DBMS This is DBMS subject	C programming This is C programming subject	WEB TECHNOLOGY This is WT Subject
			DBMS This is DBMS subject
			C programming This is C programming subject

🪟

🔍 Type here to search

📄

🖨

🌐

📁

🛒

🌐

📄

📄

📄

📄

📄

🔍

⬆

📶

ENG

US

21:03

16-08-2017

🗨



[WT](#)
[DBMS](#)
[C Prog](#)

Click on hyperlink



```
<!DOCTYPE html>
<html>
<head>
<title>HTML Frames</title>
</head>
<frameset cols="40%,60%">
  <frame name="menu" src="menu.html" />
  <frame name="main" src="main.html" />

  <noframes>
    <body>    Your browser does not support frames.  </body>
  </noframes>
</frameset>
</html>
```

Example of frames

Menu.html

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Frames</title>
</head>
<body>
<a href="page1.html" target="main">WT</a>
<br>
<a href="page2.html"
target="main">DBMS</a><br>
<a href="page3.html" target="main">C
Prog</a><br>
</html>
```

main.html

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Frames</title>
</head>
<body>
<p> Click on hyperlink </p>
</body>
</html>
```

HTML – IFRAMES

- You can define an inline frame with HTML tag <iframe>.
- The <iframe> tag is not somehow related to <frameset> tag, instead, it can appear anywhere in your document.
- The <iframe> tag defines a rectangular region within the document in which the browser can display a separate document, including scrollbars and borders.

```
<html>
<head>
<title>HTML Iframes</title>
</head>
<body>
<p>Document content goes here...</p>
<iframe src="/html/menu.htm" width="555" height="200">
  Sorry your browser does not support inline frames.
</iframe>
<p>Document content also go here...</p>
</body>
</html>
```



Document content goes here...



Document content also go here...

Attributes of Frameset and Frame

- Attributes of frameset:
 - Rows
 - Cols
 - Border
 - Frameborder
 - framespacing,
- Attributes of frame:
 - Src
 - Name
 - Frameborder "0" "1"
 - Scrolling "yes","no","auto"
 - Resize "noresize"
- Attributes of iframe:
 - Src
 - Name
 - Frameborder
 - Scrolling "yes","no","auto"
 - Resize "noresize"

- Block Elements :

- Block elements appear on the screen as if they have a line break before and after them. For example, the <p>, <h1>, <h2>, <h3>, <h4>, <h5>, <h6>, , , <dl>, <pre>, <hr />, <blockquote>, and <address> elements are all block level elements. They all start on their own new line, and anything that follows them appears on its own new line.

- Inline Elements

- Inline elements, on the other hand, can appear within sentences and do not have to appear on a new line of their own. The , <i>, <u>, , , <sup>, <sub>, <big>, <small>, , <ins>, , <code>, <cite>, <dfn>, <kbd>, and <var> elements are all inline elements.

Grouping HTML Elements

There are two important tags which we use very frequently to group various other HTML tags (i) <div> tag and (ii) tag

```
<div style = "color:red">
```

```
  <h4>This is first group</h4>
```

```
  <p>Following is a list of vegetables</p>
```

```
  <ul>
```

```
    <li>Beetroot</li>
```

```
    <li>Ginger</li>
```

```
    <li>Potato</li>
```

```
    <li>Radish</li>
```

```
  </ul>
```

```
</div>
```

```
<div style = "color:green">
```

```
  <h4>This is second group</h4>
```

```
  <p>Following is a list of fruits</p>
```

```
  <ul>
```

```
    <li>Apple</li>
```

```
    <li>Banana</li>
```

```
    <li>Mango</li>
```

```
    <li>Strawberry</li>
```

```
  </ul>
```

```
</div>
```

Following is a list of vegetables

- Beetroot
- Ginger
- Potato
- Radish

This is second group

Following is a list of fruits

- Apple
- Banana
- Mango
- Strawberry

```
<body>
```

```
<p>This is <span style = "color:red">red</span> and this is <span style  
= "color:green">green</span></p>
```

```
</body>
```

This is red and this is green

HTML – BACKGROUNDS

- By default, your webpage background is white in color.
 - Html Background with Colors
 - Html Background with Images
- `<tagName bgcolor="color_value"...>`
- **HTML Color Coding Methods**
 - Color names - You can specify color names directly like green, blue or red.
 - `<table bgcolor="red" >`
 - Hex codes - A six-digit code representing the amount of red, green, and blue that makes up the color.
 - `<table bgcolor="#ff0000" >`
 - Color decimal or percentage values - This value is specified using the `rgb()` property.
 - `<table bgcolor="rgb(255,,0,0)" >`

HTML – COLORS

- The <body> tag has following attributes which can be used to set different colors:
 - bgcolor - sets a color for the background of the page.
 - text - sets a color for the body text.
 - alink - sets a color for active links or selected links.
 - link - sets a color for linked text.
 - vlink - sets a color for visited links - that is, for linked text that you have already clicked on.

HTML Colors - Color Names










W3C Standard 16 Colors

Here is the list of W3C Standard 16 Colors names and it is recommended to use them.





	Black		Gray		Silver		White
	Yellow		Lime		Aqua		Fuchsia
	Red		Green		Blue		Purple
	Maroon		Olive		Navy		Teal

HTML Colors - Hex Codes

Each hexadecimal code will be preceded by a pound or hash sign #. Following is a list of few colors using hexadecimal notation.

Color	Color HEX
	#000000
	#FF0000
	#00FF00
	#0000FF
	#FFFF00
	#00FFFF
	#FF00FF
	#C0C0C0
	#FFFFFF

HTML Colors - RGB Values

Color	Color RGB
	rgb(0,0,0)
	rgb(255,0,0)
	rgb(0,255,0)
	rgb(0,0,255)
	rgb(255,255,0)
	rgb(0,255,255)
	rgb(255,0,255)
	rgb(192,192,192)
	rgb(255,255,255)

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Background Colors</title>
</head>
<body>
  <!-- Format 1 - Use color name -->
  <table bgcolor="yellow" width="100%">
    <tr><td> This background is yellow </td></tr>
  </table>

  <!-- Format 2 - Use hex value -->
  <table bgcolor="#ff0000" width="100%">
    <tr><td> This background is red </td></tr>
  </table>

  <!-- Format 3 - Use color value in RGB terms -->
  <table bgcolor="rgb(0,255,255)" width="100%">
    <tr><td> This background is cyan </td></tr>
  </table>
</body>
</html>
</html>
```

This background is yellow

This background is sky blue

This background is cyan

Html Background with Images

- <tagname background="Image URL"...>

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>HTML Background Images</title>
```

```
</head>
```

```
<body>
```

```
<!-- Set table background -->
```

```
<table background="/images/html.gif" width="100%" height="100">
```

```
<tr><td>
```

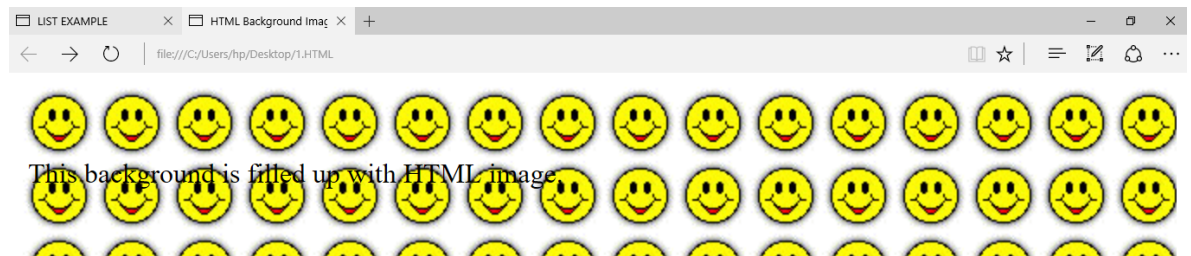
This background is filled up with HTML image.

```
</td></tr>
```

```
</table>
```

```
</body>
```

```
</html>
```



HTML – FONTS

- The font tag is having three attributes called **size, color, and face**.
- **Set Font Size:** You can set content font size using size attribute. The range of accepted values is from 1(smallest) to 7(largest). The default size of a font is 3
 - `Font size="1"
`
- **Setting Font Face :**
 - `Times New Roman
`
 - `Verdana
`
 - ` hello `
- **Setting Font Color :**
 - `This text is in pink
`
 - `This text is red`
 - ` This text is red`

The <basefont> Element:

- The <basefont> element is supposed to set a default font size, color, and typeface for any parts of the document that are not otherwise contained within a tag.
- You can use the elements to override the <basefont> settings.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Setting Basefont Color</title>
```

```
</head>
```

```
<body>
```

```
<basefont face="arial, verdana, sans-serif" size="2" color="#ff0000"> <p>This is the  
page's default font.</p>
```

```
<h2>Example of the &lt;basefont&gt; Element</h2>
```

```
<p><font size="+2" color="darkgray">
```

This is darkgray text with two sizes larger

```
</font></p>
```

```
<p><font face="courier" size="-1" color="#000000"> It is a courier font, a size smaller  
and black in color.
```

```
</font></p>
```

```
</body>
```

The <basefont> Element:



This is the page's default font.

Example of the <basefont> Element

This is darkgray text with two sizes larger

It is a courier font, a size smaller and black in color.

- Every HTML element has a default display value, depending on what type of element it is.
- There are two display values: block and inline.
- two commonly used block elements are: `<p>` and `<div>`.
- The `<p>` element defines a paragraph in an HTML document.
- The `<div>` element defines a division or a section in an HTML document

Block level tags

- <address><article><aside><blockquote><canvas><dd><div><dl><dt><fieldset><figcaption><figure><footer><form><h1>-<h6>
- <header><hr><main><nav><noscript><p><pre><section>
- <table><tfoot><video>

- An inline element does not start on a new line.
- An inline element only takes up as much width as necessary.
- <a><abbr><acronym><bdo><big>
<button>
- <cite><code><dfn><i><input><kbd><label><map>
- <object><output><q><samp><script><select><small><sub><sup><textarea><time><tt><var>
- **Note:** An inline element cannot contain a block-level element!