html => UI design/Front-end

css => changing look & feel of web page.

javascript => client side validations

course target for designing static web applications

+

advjava/asp.net/php/nodeJs

**Network:** Collection of computers interlinked together is called network. First network name is **ARPANET** (Advanced Research Projects Agency Network). First protocol in IT industry is FTP (File Transfer Protocol).

**Internet:** Internet stands for international networking.

The Internet is a network of connected computers. No company owns the Internet; it is a cooperative effort governed by a system of standards and rules. The purpose of connecting computers together, of course, is to share information.

**A Brief History of the Web**

The Web was born in a particle physics laboratory (CERN) in Geneva, Switzerland in 1989. There a computer specialist named **Tim Berners-Lee** first proposed a system of information management that used a “hypertext” process to link related documents over a network. He and his partner, **Robert Cailliau**, created a prototype and released it for review. For the first several years, web pages were text-only. It’s difficult to believe that in 1992, the world had only about 50 web servers.

**The World Wide Web Consortium**

World Wide Web Consortium (called W3C) is the organization that oversees the development of web technologies. The group was founded in 1994 by Tim Berners-Lee, the inventor of the Web, at the Massachusetts Institute of Technology (MIT).

**Tim Berners-Lee (WWW/HTTP), Cerf & Kahn (TCP/IP), Baran, Davies, Kleinrock & Roberts (packet networking), Bob Metcalfe (Ethernet).**

**WHAT IS APPLICATION OR SOFTWARE?**

Automation of manual business operations by using a programming language.

**TYPES OF APPLICATIONS OR SOFTWARES**

We can create an application or software in fallowing flavors:

1. **Desktop:** The applications which are installable in local systems are called desktop applications.
2. **Mobile:** The applications which are installable in mobile phones or tablets downloaded from play store for android and apple store for ios.
3. **Web:** The applications which are deployable in any server and can be accessible from any location using browser.

**WHAT IS WEB APPLICATION?**

Web applications are network enable applications. We can deploy any web applications in servers and we can access them over network using server ip address and application name.

In computing, a **web application** or **web app** is a client–server software **application** which the client (or user interface) runs in a **web** **browser** and it contains web documents in the form electronic pages(web pages).

**A web application typically contains fallowing three layers:**

**Presentation layer** is a user interface (views) which are accessible from any web browser.

**Business layer** is a server side program which is nothing but automation of business rules. Client layer will interact with business layer to persist data.

**Data layer** is database software where we can store client related data. Business layer will interact with data layer.

**BASIC WEB ARCHITECTURE**

**WEB BROSER**

URL: http://www.google.com

**SERVER**

HTML

JAVA/JEE

.NET

PHP

JAVASCRIPT ETC.

**DataBase**

**How the Web Works**

1. When you connect to the web, you do so via an Internet Service Provider (ISP). You type a domain name or web address into your browser to visit a site; for example: google.com, oracle.com, microsoft.com.
2. Your computer contacts a network of servers called Domain Name System (DNS)servers. These act like phonebooks; they tell your computer the IP address associated with the requested domain name. An IP address is a number of up to 12 digits separated by periods / full stops. Every device connected to the web has a unique IP address; it is like the phone number for that computer.
3. The unique number that the DNS server returns to your computer allows your browser to contact the web server that hosts the website you requested. A web server is a computer that is constantly connected to the web, and is setup especially to send web pages to users.
4. The web server then sends the page you requested back to your web browser.

**What is web browser?**

It is client side lightweight software installed in client machine. It sends http request from client to server; it takes http response from server.

Browser provides navigation among web pages, and browsers executes html, css, JavaScript files and displays output to user.

**List of Computer Browsers:**

Internet Explorer(1995),Opera(1995), Mozilla Firefox(1998), Safari(2005), Google Crome(2008) etc…3

**List of Mobile Browsers:**

Mobile Safari (iOS), Android Browser (Android), BlackBerry Browser (RIM), Nokia Browser (Symbian), Opera Mobile and Mini (installed on any device), Internet Explorer Mobile (Windows Phone), Silk (Kindle Fire) etc…

**Client:**

**It is a machine or device (desktop or laptop or tablet or mobile phone or TV etc), which can access the data from server machine.**

**The device which is used by the user is called as “Client”, person who is working on client machine is known as User.**

**Email:** Electronic mail services. It is a free service to communicate with other internet users. Email is invented by **Shabeer Bhatia.** Sabeer Bhatia is an Indian entrepreneur who founded the webmail company Hotmail.com.

**SMTP:** Simple Mail Transfer Protocol. It takes care of delivering emails from one server to another.

**MIME:** Multipurpose Internet Mail Extensions. It exchanges different kinds of data.

**Blog:** It is daily updating website or webpage. Every post displayed in reverse chronological order.

**Forum:** It is an online discussion website to exchange resources each other.

**Http:** It is a transfer protocol to exchange hypertext documents in the world wide web.

**Http(s):** Secured transfer protocol to exchange hypertext documents with the help of SSL(ciphertext).

**Ciphertext** is encrypted text. Plaintext is what you have before encryption, and **ciphertext** is the encrypted result. The term cipher is sometimes used as a synonym for **ciphertext**, but it more properly **means** the method of encryption rather than the result.

**HOW MANY TYPES OF WEB APPLICATIONS WE HAVE?**

A webpage is an electronic page developed on HTML. It is classified into two types.

Static webpage : A user unable to interact directly with these webpages. Eg: HTML, CSS

Dynamic webpage: End-user can able to interact directly with these webpages. Eg: HTML, CSS & Javascript

Collection of WebPages or web documents is called web application (website). These are classified into two types:

**STATIC WEB APPS:** The applications which can’t able to handle business logic are known as static web apps. Static apps will contain only client layer. We can develop static web applications using HTML. To provide look and feel to these static pages we can use CSS. To handle client layer business logic we can use JavaScript. We can’t able to maintain end user interaction (state) using static web apps.

Html/css/js/boot

**DYNAMIC WEB APPS:** The applications which can able to handle business logic are known as dynamic web apps. These types of apps contain at least 2 layers client and business. If we need to store client data then these application contains data layer too. We can develop client layer using HTML, CSS & javascript. Business layer using any one of the server programming language like .NET, JAVA/J2EE & PHP. We can store end user data using any database like mongo db, MS-SQL, MySql, Oracle etc.

Html/css/js/boot + advjava/asp.net/php/nodejs

**Server** 🡺 its machine, it contains data and source code of deployed web applications/sites.

* It servers data to all clients machines based request.

**Client** 🡺 its machine it sending data to server & receive data from server

**User** 🡺 user is human, how is operating client machine (means we are users, our computers are clients)

**Browser** 🡺 browser is sw, it’s a set of programs.

* Browser is a tool; it’s an interface between client and server.

**Request** 🡺 sending data from client to server machine is called as request. (input)

🡺 request should be send in the form of “HyperText”.

**Response** 🡺 sending data from server to client is called as response. (output)

🡺 the response should be send in the form of “HyperText”

**Deployment/hosting** 🡺 after completion of development & testing, we are copying total project source code from out system into main server this process is called deployment/hosting.

* Word press tool, joomula, git

**What is HTML?**

It is specially designed hypertext for web browsers, with meaningful tags or elements in simple English language.

**HTML Versions**

From W3C organization there are fallowing versions released.

**Version Specification Release Date**

**1.0 N/A (HTML 1.0) 1-Jan-1994**

**2.0 HTML 2.0 24-Nov-1995**

**3.2 W3C: HTML 3.2 14-Jan-1997**

**4.0 W3C: HTML 4.0 24-Apr-1998**

**4.1 W3C: HTML 4.1 24-Dec-1999**

**5.0 WHATWG 28-Oct-2014**

**(Adv Markup Language For Mobiles)**

**5.1 W3C: HTML 5.1 -Nov-2016**

**(Adv Markup Language For Small Electronic Devices)**

**5.2 W3C: HTML 5.2 14-Dec-2017**

**1990 => GML => SGML => HTML**

**HTML Intro**

1. HTML was developed by “Tim-Berners-Lee”, released in 1994 and maintained by W3C.Org
2. HTML stands for “Hypertext Markup Language”.
3. “Hypertext” means the text that can be transferred from internet server to internet client.

"Markup Language" means which syntax will be in the form of tags or you simply "mark up" a text document with tags that tell a Web browser how to structure it to display.

1. Technically, HTML is not a programming language, but rather a markup language.

HTML, XHTML, SHTML, XML, WML

1. HTML is used to design "**static web pages**", means HTML is used to create elements (such as headings, paragraphs, icons, menus, logos, images, textboxes, button etc) in the web pages.

static webpage means, that pages always showing same information.

1. HTML is very easy to understand (no pre-requisites).
2. HTML is “client side tech”. That means the html code executes on the client browser but not in server.

**web tech:**

which sw are supporting to design web pages or providing API to dev web coding those sw are called as web tech.

> **client side tech** ex: html/css, js, jquery, ...

Used for static web pages.

bw rec source code & trans after execution then produced the output.

> **server side tec**h ex: servlet, jsp, asp.net, php, cgi, nodejs, cold fusion ...

Dynamic web pages.

Code translation, execute with in server only, and produced

output, this output sent to client machine.

1. HTML is supported by all the browsers such as Google Chrome, Mozilla Firefox, Microsoft Internet Explorer, Safari, Opera and other browsers.
2. HTML is used in all real time web sites today, html is the only language available in world for designing webpages.
3. The file extension either "filename**.html**" or "filename**.htm**"
4. HTML is a interpreter based language. That means the HTML code will be converted into machine language in line-by-line manner. Browser interprets HTML code.

**Translators:** converting high level code (human) into machine level code (MP/OS) is called as translation. who performs this operation those are called as translators.

types: >compiler ex: c, cpp,...

>interpreter ex: html, js, oracle,...

>assembler

1. for working html no need install any software, and **browse**r is responsible for executing & producing output of html p`rograms.
2. html is 100% error f`ree language.
3. HTML is not a case sensitive language that means you can write the html code in either upper case or lower case.

**how type html & execute programs**

> open any text editor (sw) and type program.

editplus, notepad++, notepad, textpad, sublime, ms vs, word,...

>save that program with any name (.html or .htm) and anywhere in system.

but filename must be single word (space are not allowed).

> execution:

**1st Approch:** goto file location, then dbl click on file

**2nd Approch:** goto file location, then right click on file and click on open then select browser

**3rd Approch:** open any browser, then goto address bar and type filename with address.

*d:\siva\a.html*

[**Tag**](mailto:support@nareshit.comTag)**:**

* **A tag is a keyword, enclosed within "<" and ">" in HTML language.**
* **It is special kind of text placed between left angular brace and right angular brace(<tag\_name>).**
* **Tag is instruction / command given to browser.**
* **Tag is used to display some specific output in the web page.**
* **browser is not indentified the tag, it shows blank page or it prints as text.**
* **tags also represented as a elements.**
* **tag has some attributes(properties), those are used to change look & feel (components or output).**

**types of tags:**

in html we have two types tags, those are:

> **paired tags**

contains open tag and closing tag.

opening tag specifies starting point of operation/output, closing tag specifies ending point of operation/output.

Syn: **<tagname>** something **</tagname>**

**ex: <html> ... </html>**

**<head> ... </head>**

**<script> ... </script>**

**<style> ... </style>**

**<p> ... </p>**

**note: paired tags also called as body-full tags**

>**unpaired tags**

contains only open tag.

Syn: **<tagname>**

**ex: <br>**

**<img>**

**<input>**

**<hr>**

**note: Unpaired tags also called as body-less tags**

**Structure of HTML**

as per **w3c** we have to follow the following structure to design web pages (but nits not comp).

<!DOCTYPE html>

<html >

<head>

non-content sec

</head>

<body>

content sec

</body>

</html>

generally html page contains three parts, those are:

> versioning section

> head section

> body section

**versioning section**

this is providing information to browser which version we are using in webpage/program,

so browser is interpretation of code and producing result as per given specification.

Syn:

<!DOCTYPE versioning>

HTML4.0:

**<!DOCTYPE html public "-//W3C//DTD HTML 4.0//EN" "http://www.w3c.org/TR/html4/strcit.dtd">**

XHTML:

**<!DOCTYPE html public "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3c.org/TR/xhtml1/DTD/xmhtl1-strcit.dtd">**

HTML5:

**<!DOCTYPE html>**

strcit.dtd file (document type definition), it contains definitions of tags, specifications.

doctype is not case-sen, so we type in any case.

ex: DOCTYPE => valid (recommended)

doctype => valid

Doctype => valid

**html tag**

the <html> tag represents starting and ending of html program. html tag contains two child/sub tags those are head tag and body tag.

**head tag**

head tag represents non-content information (means not output) of the web page.

this information doesn't appear on webpage/in browser (it's called as non-content), but it's used internally by the browser.

this tag is used to set icons, title, to provide some meta data(info about web app), css settings, java scripting etc...

head tag contains some child/sub tags, those are

Syn:

**<head>**

**<link>**

**<title> </title>**

**<meta>**

**<style> </style>**

**<script> </script>**

**...**

**</head>**

**body tag**

body tag represents content information (means output) of the web page.

this information appear on webpage/in browser (its called as content).

this tag is used to design UI or to display output.

body tag contains so many child/sub tags.

some of tags:

<body>

<form>

<h1>

<h2>

<p>

<div>

<input>

<a>

<audio>

<vedio>

<iframe> etc...

</body>

**html is collection of tags(elements) and attributes.**

**tags are used design something or to print data on webpage.**

**every tag they had some default attributes (inbuilt values)**

**attributes are used for changing settings of tags.**

**<tagname attribute=value attribute=value... >**

**comment lines**

**comment lines are to provide some description abt of our program.**

**Syn:**

**<!-- comments -->**

**comment are not executed by browser.**

**Title tag**

>”title” tag used to set title/caption of web page, it displays in browser’s title bar.

> it used paired

> “title” tag should be used in “head” tag.

**Syn: <title> Text </title>**

**How to set icon to a web page?**

Using “link” tag we can set icon to our web page.

“link” is un-paired tag.

“link” tag should be used in “head” tag.

**Syn:- <link rel=”icon” href=”filename”>**

**heading tags**

these tags are used to print data/text in different sizes.

html providing 6 heading tags, those are h1, h2, h3, h4, h5, h6.

six tags are paired tags.

**Syn:**

**<h1> text </h1>**

**<h2> text </h2>**

**<h3> text </h3>**

**<h4> text </h4>**

**<h5> text </h5>**

**<h6> text </h6>**

**p tag**

> p stands for paragraph.

> this tag is used to display/print more lines of text (paragraph)

> its paired tag.

> browser display an empty line(gap) between paragraphs (5px to 10px)

Syn:

**<p> text or info </p>**

**Note:**

**>**browser/html doesn't accept more than one space, means while designing of program we given more space but browser prints only one space.

**>**browser/html doesn't accepts enter key (line breaking), means while designing of program we use enter key but browser prints not breaking line.

**br tag**

* br tag used for line breaking (enter key)
* br is unpaired

**Syn: <br>**

**Html special chars**

>these special chars are used to produce some special output, like space, $, euro, pound, PI, ¼, ½, ¾ , less then, greater than etc…

>special char should start with &

**Syn: &spe-char;**

**Formatting tags**

**b tag**

> b stand for bold

> b tag used to print text in bold format

> b is paired tag

**Syn:**

**<b> text </b>**

**u tag**

> u stand for underline

> u tag used to print text with underline (draws a line base of text)

> u is paired tag

**Syn:**

**<u> text </u>**

**i tag**

> i stand for italic (inclined)

> i tag used to print text with little banding

> i is paired

**Syn:**

**<i> text </i>**

**strikeout tag**

> strikeout tag used to print text with line (draws a line middle of text)

> strikeout is paired tag

Syn:

<strike> text </strike>

**superscript tag**

>this tag used to display text top of upper line

> superscript is paired tag

Syn:

<sup> text </sup>

**subscript tag**

>this tag used to display text bottom of baseline

> subscript is paired tag

Syn:

<sub> text </sub>

**nbsp**

nbsp stands for non-remove blank space

Syn: &nbsp;

**br tag**

> br used for line breaking (enter key or \n)

> br is un-paired (Single)

Syn:

<br>

**Note:** in html tab, space bar, enter key are not accepted (browser not processing these keys)

**pre tag**

> pre stands for pre-formatting

> pre tag is used to print data/text, how we typed in same format in browser.

> pre is paired tag

**Syn:**

**<pre> text </pre>**

**attributes**

> attribute is a special feature/setting/property of a tag.

> attributes are used to change the default look/feel of data or element.

> every tag they have attributes(1-n)

Syn:

**<tagname attribute="parameter" attribute="parameter" ...>**

parameter means the values of attribute.

**Types:**

as per **html5** we have 4types of attributes, those are:

>global attributes (universal)

>specific attributes (personal)

>event attributes (dynamic)

>optional attribute <== html5

**global attributes:**

these attribute are common for most of tags (99% of tags)

those attributes are:

class, id, name, style etc...

ex:

<h1 class="" id="" name="a" style="" ...>

<p class="" id="" name="b" style="" ...>

<pre class="" id="" name="c" style="" ...>

**specific attributes:**

these attribute are specific to some tags/elements only (not common).

those attributes are:

src, href, action, selected, width, height, etc...

ex:

<a href="url" ... >

<img src="" ...>

<form action=”url”>

**event attributes:**

these attribute are used to some logical operations on tags/elements.

logical operations we can perform by using JavaScript, these also called dynamic attribute.

attributes are:

onclick, oninput, onfocus, onexit, onload, onchange, ....

ex:

<button onclick="js code/fun1">

<body onload="js code">

**optional attributes:**

these attribute are not comp to specify/to use.

these type attributes are supported since html5.0.

those attributes are:

type, method ...

ex:

<style type="text/stylesheet" ...>

<script type="text/javascirpt" ...>

<link type="image/jpg" ...>

<form method="get" ...>

**categories of attributes:**

html attributes <tag attribute="value" attribute="value" >

css attributes <tag style="css-attribute:value; css-attribute:value;...">

**note:**

css attributes we can't use in place of html attributes.

html attributes we can't in place of css attributes.

**images**

> "img" tag is used to display images on webpage.

> in one web pages we can display any no.of images and any type of images.

> it is strongly recommend to all images in side root folder or create sub folder with name images in root folder

> its un-paired tag

Syn:

**<img attributes>**

**attributes:**

src => to specify which img you want to display (must use)

width => width of image (pixel)

height => height of image (pixel)

title => it is used to specify tool tip. (whenever mouse pointer comes on top of image)

alt => alternative text, if image not loaded in webpage/not display, we want to display text message to user it called as alt

border => it draws a line around the image

+

global attributes

**hyperlinks**

> "a" tag is used to create a hyperlink, whenever user clicks on the hyperlink, the specified page is open.

> destination page sometime within same application or other application.

> a stands for "anchor"

>web application basically contains links to other pages, so it's very common used tag.

> by default every browser provides built-in style for each hyperlink,

i.e blue color+hand symbol+under line.

> we can customize these style by using CSS.

> its paired tag

**Syn:**

<a attributes>Display Text</a>

<a attributes> <img> </a>

**attributes:**

href : hyper reference, used to specify the address of webpage or web site, i.e whenever user clicks on this link, which page you want to open specify that url.

url may be html page, server side file, image, audio file, video, pdf file, documents etc...

target : where you want open destination page

\_blank ==> opens the link in a window/tab

\_self ==> opens the link in current working tab/window (its default)

\_parent ==> opens the link in parent frame

\_top ==> opens the link in full body of window

framename ==> opens the link in specified frame

**html colors**

html supports 3types of patterns, those are

> named colors

> RGB colors

> Hexadecimal number colors

**named colors:**

>it sup to write direct color name

>we have some limited colors (15 to 20)

ex: white, black, red, green etc...

> color names are not case-sen

**RGB colors:**

>RGB model specifies that the composition of 3 basic colors (Red, Green, Blue)

>RGB produces 16millions colors.

Syn: rgb(red,green,blue)

red => 0 - 255

green => 0 - 255

blue => 0 - 255

ex: rgb(10, 45, 120)

**Hexadecimal number colors:**

>Hexadecimal model is the shortcut for rgb model

>Hexadecimal system range from 0 - 15

0,1,2,3,4,5,6,7,8,9,a,b,c,d,e,f

Syn: #redgreenblue

ex: #11bb4e

**Note:** in real time "Hexadecimal model" is recommended.

these colors we can use for foreground color, background color, border color etc..

for setting colors we have some attributes, those are

color

background-color

border-color

all these are CSS attributes.

**How to set multiple colors as a background?**

Html/css both supports set multiple colors as a background its called as gradient.

Using linear-gradient() we set colors.

**working with list tags**

these tags are used to display data/info in points wise.

html supports three types of list, those are

> ol (numbering style)

> ul (bulleting style)

> dl

ol tag

>ol stands for "Ordered List".

>it is used to display the list of items(names, colors, team names, course name...) with numbering.

>it supports 5types numbering, those are 1, A, a, i, I. by default it displaying in number.

> by using "ol" tag we can create ordered list

>ol is paired tag

**li tag**

> li stands for "list item"

> li is sub tag of ol tag

> li tag is used to create items for list

> li is paired tag

Syn:

<ol attributes>

<li> text </li>

<li> text </li>

<li> text </li>

...

</ol>

**ol attributes:**

type : which type numbering to display (Default is number)

start : from where u want to start numbering (default is 1)

reversed : to displaying items in desc order

**li attributes:**

value : used for restarting numbering with specified value

Syn: <li value=”text”>

**ul tag**

>ul stands for "Un-Ordered List".

>it is used to display the list of items(names, colors, team names, course name...) with bulleting.

>it supports 3types bulleting; those are **dot, circle, square**, by default is dot.

>by using "ul" tag we can create un-ordered list items

> ul is paired tag

>"li" tag used for creating list items

**Syn:**

<ul type="dot/circle/square">

<li> text </li>

<li> text </li>

<li> text </li>

...

</ul>

**dl tag**

>dl stands for Definition list (since html5 description list)

>dl tag used for to display definitions/full forms (collection of definitions)

>its paired tag

> "dt" and "dd" are sub tags of "dl" tag

> "dt" stands for definition title, "dd" stands for definition data.

> dt & dd are paired

Syn:

<dl>

<dt>title/word</dt>

<dd>information</dd>

<dt>title/word</dt>

<dd>information</dd>

<dt>title/word</dt>

<dd>information</dd>

...

</dl>

**fieldset tag**

> this tag used for drawing a line/border around elements/controls.

> its paired tag

> we can draw any no.of borders

**Syn:** **<fieldset attributes>**

**designing/text**

**</fieldtset>**

**attributes:**

align : align of elements, it supports 3 alignments center, left, right

left is default align

border : style of line, thickness of line, color of line

width : width of box (size in % )

**legend tag**

>legend tag used for set title/heading for fieldset

>legend is sub tag of fieldset tag

>its paired tag

**Syn:** <legend attributes>Heading</legend>

**attributes:**

align :align of elements, it supports 3 alignments center, left, right

left is default align

color :

**table tag**

>table tag is used to display the data in form rows & cols in the web page.

> a table is a collection of rows, each row is collection of cells/col/field.

> a table is represented as <table> tag, a row represented as <tr> tag, a colheading is represented as <th> tag, data rep as <td> tag.

> table heading is represented as <caption> tag.

> <thead> tag is rep of table head and <tbody> tag is rep of table body.

> these 2tags ava since html5.0

tr => table row

th => table header(col heading)

td => table data (col data)

thead => table head

tbody => table body

caption => table caption

> all these 7tags are paired tags

**Syn:**

<table>

<tr>

<th> heading1 </th> <th>headaing2 </th>

</tr>

<tr>

<td> data </td> <td>data </td>

</tr>

<tr>

<td> data </td> <td>data </td>

</tr>

...

</table>

**NOte:**

<th> and <td> are sub tags of <tr>

<tr> is sub tag of <table>

**table attributes:**

border : border of table (0 means no border, 1-n border req)

align : alignment of table

width : width of table (%)

...

**th & td attributes:**

colspan : specifies the no.of columns to merge

rowspan : specifies the no.of rows to merge

...

To become a web developer to required HTML, CSS, JavaScript, Bootstrap

They are availavle versions...........

HTML 4V

HTML 5V

CSS 2V

CSS 3V

JavaScript 5V

JavaScript 6V

JavaScript 7V

Bootstrap 4V

Client side(Angular JS)-----------------------server side(JAVA/not understand html)

one organization-----W3C(WWW Consortium)(parepare standard for rules for internet browsres, servers)

HTML(hyper text markup language)(developed by "Tim Berners Lee")

=========================

\*hyper text means a text representation that can contains refernces to another text.

\*A document can have references to another document which is able to hyperlink

\*Whenever webpage can contains references to another webpage that text behind this webpage is called as hypertext

\*"markup language" means contains tag some particular word mentioned inside < and >

\*"markup language" which syntax will be in the form of tags.

HTML can be executed by browser

browser:-

It is a s/w that can read, understand, execute html also display output(web page) to the user.

HTML Versions:-

HTML 1.0 : NOV 1991

HTML 2.0 : MAR 1995

HTML 3.0 : JAN1997

HTML 4.01 : DEC 1997

HTML 5.0 : OCT 2014

HTML 5.1 : NOV 2016

HTML 5.2 : DEC 2017

List of imp browsers:-

1.Google Chrome

2.Mozila Firefox

3.Safari

4.Opera

5.Microsoft Internet Explorer(MSIE) [discontinued]

6.Microsoft Edge

Browsers are required to run web page and to show to the user.

File extentions:

html lang .html

css lang .css

javascript lang .js

\*tag is command/instrucions is nothing but browser to display some specific output.