Types of Web Pages:

1. i) Static → HTML, CSS
   ii) responsive → CSS, BOOTSTRAP
   iii) Dynamic → JS → Javascript (ES6)

HTML is a standard Markup Language
Tag

HTML → HyperText Markup Language.

Extention → .html

HTML Elements are building a blocks of HTML Pages.

HTML Elements are represented by Tags

USES OF HTML

*) Web document Creation
*) Internet Browing with Ease
*) Game development usage
*) Web Pages development
*) offline storage
*) User friendly.

HISTORY

Current version of HTML is [HTML5, CSS3]

Browsers → Edge, chrome, opera,
Platform → Windows11, Linux, mac, Android.
Text editors → Notepad, Notepad++, visual studio code

× How to run HTML program.

Visual studio code

✓ Alt + F ⇒ open File menu
✓ Alt + E ⇒ open Edit Menu
Alt + S ⇒ open Selection.
Alt + V ⇒ open view
Alt + g ⇒ open Go
✓ Alt + R ⇒ Run Menu

Ctrl + N ⇒ New File

⇒ Toggle icon.

Tag elements → container elements → opening & closing tag
↳ Stand alone elements.
↳ only opening tag

"Clean-coded Format" → HTML code should be in this format

↳ Folder structure
↳ task name
↳ File and folder should be in proper name
↳ avoid unwanted space
↳ Description is Mandatory.

HTML is an "Non-case sensitive" (closing tag, capital are not necessary. But it is recommended to follow the proper way)

*) How to run an HTML File:

① Run Menu → Run without Debugging → web app (chrome)

Run with "Live Server": ② Live Server Method.

Go live:

Default port number 5500. [127.0.0.1:5500]

③ copy path: [shift + Alt + C]

Right click on File name, select "copy path".

* Background color:

① RGB → Red, Green, Blue.

② CMYK → Cyan, Magenta, Yellow, Black (Key)

*) Elements:

Inside of the tag is called "Elements".

*) How to change Background colour:

<body bgcolor = "red"

<body bgcolor = "#1E02EE"

~~bgcolor = "red".~~

All HTML files should show both "output" and "Dev tools".

In the Browser (File) → right click → inspect

what are Chrome developer Tools:

i) Elements
ii) Console
iii) Source
iv) Network
v) Application
vi) Security
vii) Memory
viii) Performance
ix) Audits.

i) **Element** shows you the HTML used to build the page you're looking at, together with any inline CSS. (only codes)

ii] **Console** → To find the errors (Javascript output).
↳ Two types of windows. → REPL, CLI
Read Eval Print Loop
↳ command Line Interface

Loop → repeated work

iii] **Source** tab → tab shows you where all the files that were used to make a website are stored and let you inspect them.
local files.

iv] **Network** tabs shows you all the files that are loading in the URL you're looking at.

v] "Application" shows you what's in your browser Storage : in browser databases like mongodb, mysql

vi) Security : security gives you information, letting you view a site's HTTPS certificate and TLS status.

HTTPS → Hyper Text Transfer Protocol Secure

TLS → Transport Layer Security.

SSL → Secure Soket Layer

For (S) the certification needed →

What is HTML 4 :

→ It is a markup language, that published in 1997 as a W3C Recommendation.

→ HTML 4 is the extented version of HTML 3.2.

Latest form of HTML is HTML5 version, that is a www Consortium (W3C) recommendation. It was released on 22nd Jan 2008.

Structure of HTML5

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title> Project Title </title>
    <meta charset="UTF-8">
  </head>
 <body>

 </body>
 </html>
```

HTML → root.

<!Doctype html> helps browsers to understand the version of html the document is written in for better interpretability. Doctument type.

*) <html lang = "en">. it specifies the language of the element content.

Example : "en" for english, "es" for Spanish, "fr" for french.

*) <head> contains the title of the page and information of the page.

*) <Meta charset = "utf-8">
  *) Introduced in HTML5.
  *) meta represent the web pages
  *) charset is an "attribute" character set.
    ↳ used for (encoding purposes)
  *) ~~uniform tr~~ utf => Unicode transform Format

*) <title> tag defines the title of the project.

*) <body> contains the body of the actual contents of the page.

*) <heading :

Default size of h1 = 32px (pixels)
h2 = 24px
h3 = 20.8px
h4 = 16px
h5 = 12.8px
h6 = 11.2px

<h1> This is heading 1 </h1>

<Starting tag attribute_name = "AttributeValue" > content </closing tag>

"Property: value;".

<h1 style = "Font-size : 50px;"> Heading 1 </h1>

Paragraph tags:

Syntax : <P> contents </P>

lorem → Default Paragraph

Alt + Z → modify in

 is an Stand alone elements.

[react only accepts container elements]

 → void element [Stand alone element is converted into container element called void element]

<hr> = horizontal inline. (To draw a horizontal line)

---

image tag:

(image) — src: source (attribute name) — alt: alternate name (attribute name) — height: (y axis) vertical (attribute name) — width: (x-axis) horizontal (attribute name)

Go to browser → Search image → choose image → right click → open image New tab → copy the image address in address bar → Paste it in "src" (link)

if path is wrong, alt will execute. alt = "no image"

Used in local image / downloaded image.

./ → current folder, Path, directory.

Link tag:

 click here

Button: <button> click here </button>

clickhere

Button as a link:

```
<button type="button">
    <a href="https://www.google.com" target="_blank">
    clickhere </a>
</button>.
```

Image as a link:

```
<img src="./assets/nature.jpg
<a href="https://img.fu - - - -.jpg" target="_blank">
<img src="./assets/nature.jpg" alt="no image" height="100pe"
width ="100pe"/>
</a>
```

Comment:

```
<!-- Any text as comment -->
```

```
<button type="button">
    <a href="https://www.google.com" target="_blank">
    Click here </a>
</button>.
```

Image as a link:

```
<img src="./assets/nature.jpg
<a href="https://img.fu...jpg" target="_blank">
<img src="./assets/nature.jpg" alt="no image" height="100pe"
width="100pe" />
</a>
```

Comment: → single line Comments.
→ multi line

```
<!-- Any text as comment -->
```

4/6/24

Formatting tags:

1) bold tag :
2) strong :   } same
3) italic :   ,
4) underline : <u> </u> , <ins> </ins>
5) delete : <del> word <del> i.e: Strikes the word in Paragraph
6) Highlight : <mark> </mark>
7) Preformatted text : <pre> </pre>
   ↳ prints in the same format what we enter in the vscode

8) Subscript:
    H ₂ O → H_2O
9) Superscript: 1 st floor

10) abbreviation :

 <abbr title = "world wide web"> www </abbr>

11) bi-directional over ride

bdo . dir "rtl" > welcome </bdo>

↳ right to left

12) KBD = keyboard in a Document.

<kbd> ctrl + p </kbd>

<kbd> ctrl + s </kbd>

13) codes with ~~of~~ QUOTES :

<q> All is well </q>

14) code :

<code> </code>

15) 'Span' element is used to color the text".

  ↳ different font

16) address :

<address> </address>

List : unordered list, ordered list. <ol

(list) (Bullet points) (numbering)

ul types : circle, square, and disc format & none

disc is default

```
<ul type = "none">
    <li>    </li>
    <li>    </li>
</ul>
```

---

ordered list :

upper roman → I

lower roman → i

number 1 → default

upper alpha = A

lower alpha = a

Start ⇒ attribute.

<ol type = "1" start = "50"

```
<ol type = "I">
    <li>    </li>
    <li>    </li>
</ol>
```

dt => description term

dd => description data.

dl -> description list

```
<dl>
<dt> front end language </dt>
<dd> HTML5, CSS, JS </dd>
</dl>
```

Elements:

i) semantic Elements

ii) non-semantic Elements.

what are semantic Elements?

A semantic elements cleasly describes it's meaning to both the browser and the developer

<section>

<article>

<header>

<footer>

<nav>

<aside>

what are non-semantic Elements?

Non-semantic elements generally tells nothing match about it's contents, traditionally developers have implemented non-semantic elements with a class attribute to define a structure and express the meaning of content

i.e: -> division tag, non-semantic tag..

"Ctrl + 0" is the default zoom size in chrome browser

```
<header>
    <center> <h1> Resume </h1> </center>
</header>

<nav>
  <ul>
    <li> Home </li>
    <li> Skills </li>
    <li> Certifications </li>
    <li> Qualification </li>
</nav> </ul>
```

header: It specifies for a For a section or document, it should be a container of a introductic content.

Navigation: Identify a collection of navigation link, not every navigation link needs to be contained with in a nanc tag. as it is intended for primary navigation link.

Section: Identify a portion of cluster of content usually marked by a heading, then article specify an independent, self containing tag. (self contained content) i.e: block post, forum post, unsued ~~Paper~~ article.

aside: Specifies a part of content, placed aside from

Figure and Figcaption:

It is important to add the image inside the "figure tag."

Details and summary:

```
<details>
  <summary> project Title </summary>
    <p> Education website </p>
    <p> Online Food Order </p>
</details>
```

Table tag:

S.No	Name	Age	
<table>			
<thead> <tr>	<th> S.No </th>	<th> Name </th>	</tr> </thead>
<tbody> <tr>	<td> 1 </td>	<td> Dsb </td>	</tr>
<tr>	<td> 2 </td>	<td> Dhaanush </td>	</tr>
<tr>	<td> 3 </td>	<td> John </td>	</tr>
</table>			

i) <caption> : tag used for caption for a table.

ii) Merge: (use inside the td [tabledata]).

To merge 2 or more rows,

```
<td rowspan="2">
```

To merge 2 or more columns,

```
<td colspan="2">
```

*) Table border:

```
<table border="2px" style="border-collapse: collapse;">
```

border → to set border for the table
style → attribute name
border-collapse → property
collapse → value

*) Table size:

Units: Absolute units → px, Points, cm, mm.
Relative units → percentage %
em → element
rem → root element
vh → viewport height.
vw → viewport web

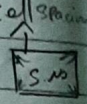
- Table colors:

RGB → Red Green Blue
RGBA → Red, Green, Blue, Alpha
HSL → Hue, Saturation, Lightness
HSLA → " " ", Alpha
HEXADECIMEL COLORCODE.

Color: → To change a text
background color: → To change the background.

* Table Padding and Spacing:

Padding: content in the box (spacing)



x) Adding images in horizontal into table:

```
<table>
  <tbody>
    <tr>
      <td>
        <figure>
          <image src="" alt="" height="" width="" />
        </figure>
      </td>
      <td>
        <figure>
          <img src="" ——— >
        </figure>
      </td>
    </tr>
  </tbody>
</table>
```

*) Form Tag:

HTML Forms are required, when you want to Collect details from "site visitor"

elements are;

- text fields
- textarea fields
- drop-down menus
- radio buttons
- checkboxes

input types:

where user can input the data.

<input type = "button" >
<input type = "checkbox" >
<input type = "password" >
<input type = "email" >
<input type = "reset" >
<input type = "file" >
<input type = "radio" >
<input type = "color" >
<input type = "text" >
<input type = "date" >
<input type = "time" >
<input type = "Submit" >

Syntax:

<input type = "text" value = " " placeholder = "Enter your name" id = "username" name = "username" />

id and name attribute is used to access the HTML element in JS page.

size = " " is used to increase the size of input box.

readonly, disabled, are other useful attributes.

"required" are important type of attributes.

<input size = "80" maxlength = "10" />

```
<table>
  <tr>
  <td>
    <label for="username">User Name</label>
  </td>
  <td>
    <input type="text" value="" placeholder="Enter username"
    id="username" name>
  </td>
  </tr>
  </table>
</form>
```

Checkbox: `<input type="checkbox" value="HTML">HTML`

radio: `<input type="radio" name="gender" /> Female`
`<input type="radio" name="gender" /> Male`

Select and option:

```
<select>
  <option>HTML</option>
  <option>CSS</option>
  <option>Bootstrap</option>
  <option>JS</option>
</select>
```

textarea:

```
<textarea rows="10" cols="20">

</textarea>
```

<fieldset>
<legend> </legend>

</fieldset>

* Alert:

<input type="button" value="Login" onclick="alert('Login success')"/>

*) Depricated tags: <b>, <italic>, <underline>, <mark>, <delete>

GLOBAL ATTRIBUTE

1. HTML contenteditable Attribute:

 This is a paragraph. It is editable. Try to change text

Definition & usage:
This specifies whether the content of an element is editable or not.

Note: when the contenteditable attribute

ie:
 Paragraph sentences can be edited

2. Add Favicon: (Important: link tag used for adding fav icon & CSS stylesheet file)
we use HTML <link> tag to attach a favicon to the document <head>

To set icon at website ← <link rel="shortcut icon" href="/images/favicon.ico" >

~~<head>~~
<title> Project Title </title>
</head>

Pre-loaded assets => Favicon, Stylesheets (CSS).

<link rel = "shortcut icon/name" href = "link" />

rel → type of document linked
href → image/icon link.

Iframe → inline Frame

<iframe
id = "inlineframeexample"
title = "Inline Frame Example"
width = "300 px"
height = "200 px"
marginheight = "40"
frameborder = "0".
src = "./assets/nature.jpg" >
</iframe>

Newly used in iframe tag

"iframe" is an "embedded webpage"

* Frameborder is used to specify whether or not to display a border around the content of an iframe element.

* Void element is an element in html that can't have any child node. & self-closing tag.
example:
area, base, br, col, command, embed, hr, img, input, keygen, link, meta, param, source, track, wbr.

* Empty elements has no child elements and no enclosing tag.

default form submission is "Get" method.

<samp> :

This tage is used to define a sample output from a computer program.

 <samp> File not found,
 Press F1 to continue </samp>

API → Application Programming interface.

Api's are using the set of instructions & Protocals

*) The Method="" attribute specifies how to send form-data (the form-data is sent to the page specified in the action attribute).

*) The form-data can be sent as a Url Uniform Resource Locator variable with method get or as a http post transaction, the method POST.

~~<form html~~ ↑ To which website to transform after submitting the form.

<form action = "./home.html" method = "post">
<fieldset>
<legend> login form </legend>
<label for = "user" > UserName </label>
<input type = "text" id = "user" name = "user" value = ""
Placeholder = "Enter a username"/>
<label for = "Pass" > password </label>
<input type = "password" id = "Pass" name = "Pass" value = ""
Placeholder = "Enter a Password"/>
<input type = "Submit" value = "submit"/>
</fieldset>
</form>.

Attribute applies the keyboard shorcut to the current element is called "access key".

with the method "Post" Data is included from body when sent to the server

with the method "get" Data is goes with URL

What is Network?

Network is used to communicate with Devices

Communication can be done by wired or wireless mediums.

wired => Data cable, Ethernet, Lan
Wireless => wifi, bluetooth

Types of Network?

within the company LAN -> local Area network
Internet Global connectivity WAN -> wide Area Network
network across city or town. MAN -> Metropolitant Area Network

Network Topology:

*) BUS Topology -> It works mainly based on the backbone link.
*) Link Topology
*) Star Topology
*) Mesh Topology.

*) Bus: if backbone is slow, entire network will be slow (only one Network as back bone.

*) Link: round (ring) All devices are connected in the circle format, all device are connected every devices has 2 neighbor.

*) Star:- All devices are connected to the central device (Hub)

*) Mesh : All devices are connected to all other devices. This Topology is more reliable, But implementation is more expensive.

Summary 6/6/24

i) contenteditable, Favicon, <link/>
ii) iframe
iii) <form action="..." method="..." >
iv) network → payload.

HTML Entities:

Html Entities are crucial Parts of HTML Markup language.
They enable you to display characters that are reserved in HTML or that or not readily available on the keyboard.

What are the HTML Entities:

Entities are used to represent special characters in the format that the browser can understand.
They start with an ampersand (&) and ends with an semicolon(;)

Uses:

Reserved characters:
Special symbols:
non-breaking space:

< for <
> for >
& for &
  for a non breaking space
© for ©

HTML Block level and Inline elements:

∴ Block level elements starts with New line
, ~~block~~ <p>, <h1>, <h2>.

∴ Inline elements does not starts with New line.
<a>, , <img.

The block-level element full with available and has a line break, before and after this.

Inline elements take it has only much width as necessary, doesn't force line break after it.

~~SVG~~ →

Designing => i) Vectors → coreldraw, illustrator.
ii) Pixel → adobe photoshop.

i) Vector: No Quality loss.

SVG → Scalable Vector Graphics.

SVG is an Markup Language that describes "two dimensional" Vector graphics.

Vector (graphics) images such as svg format, can be resized without losing quality.

SVG files can be dynamically localized using Proper library.

they also be localized without requiring the use of the graphics editor, unlike bitmapped images, such as JPEG and PNG.

Draw a circle using SVG:

r : Defines radius of circle.
cx : Indicates the position of the circle's center in the x direction.
cy : Indicates the position of the circle's center in the y direction.

```
<svg height="100" width="100">

  <circle cx="50" cy="50" r="40" fill="blue" />
          stroke="" (border)

</svg>
```

Advantages of SVG:

- Reduced file size
- Indexable
- Easily Editable
- ~~Easily c~~ Ease of scaling.

What is Canvas?

Canvas element serves as a container for graphics which can be Rendered via a scripting

Why use Canvas?

Graphics:
Dynamic content.
Interactivity:

<canvas id = "mycanvas" style = "border : 1px solid red;
width = "200" height = "100" > </canvas>

Differences between <svg> and <canvas>

	svg	canvas (workspace)
	Vector graphics	Raster graphics
	more scalable	not scalable
	can be modified.	changes can only made through scripts

*) Pattern attribute.

pattern = "[A-Z a-z 0-9] {10}"

*) <video> Tag- is used to embed video files in HTML Document.

It supports multiple attributes.

Syntax: <video src = "videoname" controls > </video>

Attributes for <video> Tag :

src : specifies the Path to the video file

controls : Adds video controls, like Play, Pause and volume.

autoplay : Automatically starts playing the video when the page loads.

loop : Repeats the video once it ends.

Muted : Mutes the video by default.

Poster: Specifies an image to be displayed before the video starts playing.

width and Height : Specifies the dimensions of the video.

*) <source> tag:

```
<video controls width="250">
    <source src="/media/cc0-video/flower.mp4" type="video/mp4"/>
</video>
```

∴ Source tag is used to video, Audio, ~~and~~ Picture.

*) <audio> tag:

```
<audio controls width="250">
    <source src="/media/...flower.mp3" type="audio/mp3">
</audio>
```

browser chooses the first supported format to play with browser's default controls.

what the "poster" attribute do in '<video' tag?

It specifies an image that

<track> tag:

The <track> element is used to child of the media elements. <audio> <video>.

Specified time and text tracks and time based data.

Tracks are formatted in "webVTT" format.
↳web video Text Tracks

Purposes:

*) The <track> tag is used for specifying subtitles and caption other types of time based text.

*) It is typically applied as a time of audio and video.