WAD UNIT 1: INTRODUCTION TO WEB TECHNOLOGIES

HTML - Hypertext Markup Language.
It is a standard markup language for creating web pages.
It provides structure & content of web pages.
It uses tags to define elements (like headings, paragraph & links).

Why HTML?

1) Structure & organization - provides backbone of web page & organizes contents.
2) Platform independence - can be displayed on any device or OS.
3) Ease of use - simple syntax with readable tags.
4) Works with other Technologies - easily integrates with CSS, JS and frameworks like Bootstrap.
5) Hypertext links - Enables creation of links connecting web pages.
6) Supports Multimedia - can embed images, videos and audio.
7) Browser friendly - HTML is supported by all browsers (Google, Opera, etc)

HTML Tags

Define the structure and content of a webpage.
Tags are instructions that are embeded directly into the text of HTML docs.
Enclosed in angle brackets < >
2 types of tags:
Paired Tags - comes in pair, with opening tag (<tagname>) and closing tag (</tagname>). Eg. This is para
Unpaired / self closing Tags - these do not require a closing tag.
Eg. <img>,
, <hr>, etc.

Examples of HTML tags.

<html>	Defines the root of HTML document.
<head>	Contains metadata, title, links to CSS
<title>	Sets the title of webpage.
<body>	Contains visible content of the page.

```
<html>
<head>
    <title> My Web Page </title>
</head>
<body>
    <h1> Welcome! </h1>
    <p> A simple web page </p>
</body>
</html>
```

<h1> to <h6>	Headings (H1 is largest, H6 is smallest)	Text-formatting Tags.
	Defines paragraph.	
<b>	Makes text bold.	
<i>	Makes text italic.	
<u>	Underlines text.	
 	Inserts a line break (self-closing).	
<hr>	Inserts a horizontal line (self-closing)	
<a>	Creates hyperlink.	Links & media tags
<img>	Embeds an image.	
<audio>	Embeds an audio file.	
<video>	Embeds a video file.	

HTML Elements :- Complete structure consisting of opening tag, content & closing tag.

Types : Normal Elements - have both start & end tags. (eg)
Void Elements - does not have closing tag. (eg <img>)

Relation betn Tags & Elements
Tags indicate start and end of elements.
Elements contain Tags & contents betn them.

Eg.
 Paragraph text
<img src = 'image.jpg' >

HTML Attributes :- Added to tags to provide more info about how tag should appear or behave.
Always specified within opening tag as name-value pairs, separated by equals sign (=).

Syntax : <tagname attribute = "value"> content </tagname>

id	Assigns unique ID to an element.	<p id = 'intro"> Hello
class	Groups elements with same class name.	Hello
style	Adds CSS styling.	Hello
title	Shows tooltip on hover.	Hover me
src	Specifies source img/media file.	<img src = 'img.jpg' alt = "An eg img">

Properties :-

Attributes vs Properties	Defined by HTML	Exist before page loads :	Remain same or unedited	<input value = "John"> default value
	Defined by DOM (Document Object Model)	Exist after page loads (inside DOM)	Can be update anytime via JS	document.getElementById ("myInput").value = "Davi"; changed the value by JS.

Headings List :- 6 levels of headings

<h1> Main Heading </h1>
<h2> Section Heading </h2>
<h3> Subsection Heading </h3>
<h4> Subsubsection Heading </h4>
<h5> Subsubsubsection Heading </h5>
<h6> lowest level Heading </h6>

Links :- links are created using '<a>' tag. Allows users to navigate betn web pages.
Has 3 parts : a destination, a label & a target.
eg. label
↳ destination
↳ will appear underlined / highlighted on the page.

Tables :- defined using '<table>' tag. organizes data into rows & columns.
They represent tabular data in structured format.

<tr> create table rows
<td> create data cells.
<th> table heading.

eg.
```
<table border = "1,">
  <tr>
    <th> Name </th> <th> Age </th>
  </tr>
  <tr>
    <td> Alice </td> <td> 25 </td>
  </tr>
</table>
```

Images :- using '<img>' tag.
Shows visual content on web pg

eg. <img src = " img.jpg"
alt = "An eg image" >

Media :— supports audio & video using <audio> & <video> tags.

eg. <audio controls>
<source src="music.mp3" type="audio/mpeg">
</audio>

eg. <video width='400' controls>
<source src="video.mp4" type="video/mp4">
</video>

HTML Form :— Required, when you collect some data from the user.

Syntax: <form action="Script URL" method="GET|POST"> </form>
action → sends data to a server
method → used to send data securely.

eg: <form action="submit.php" method="POST">
<label> Name: </label>
<input type="text" placeholder="Enter your name">
<label> Email: </label>
<input type="email" placeholder="Enter your email">
<button type="submit"> submit </button>
</form>

7 Types of form controls :—

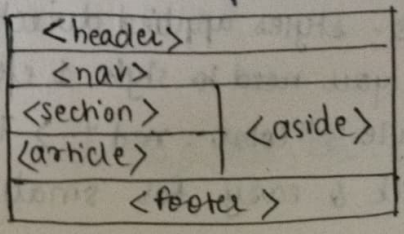
Form Control	Description	Key features	Example code
Text input	Allows user to enter single line txt	Used for names, usernames, etc	<input type="text" placeholder="Enter y nm">
Password i/p	Similar to txt but hides entered char.	Used for sensitive data.	<input type="password" placeholder="Enter password">
Email input	Accepts only valid email address.	Ensures proper format	<input type="email" placeholder="Enter email">
Number i/p	Accepts only numeric values.	can have max/min values.	<input type="number" min="1" max="100">
Checkbox	Allows multiple selections from given options.	Users can check or uncheck multiple boxes.	<input type="checkbox"> Subscribe.
Radio Button	Allows only 1 selection from a grp.	Requires name attribute for grouping.	<input type="radio" name="gender" value="male"> Male.

Select | File upload | Hidden form | Button ① Text Vs Password ② Checkbox Vs Radio inp

Semantic HTML 5 Elements :— semantic elements clearly describe their content & purpose.

<header>	defines header section
<nav>	set of navigation links
<main>	main content of webpg
<figure>	img, diagrams
<footer>	bottom/foot of page

<time> represent time



CSS - Cascading Style Sheets

Adds style to web documents (styles HTML elements).

It controls the layout, colors, fonts, spacing, animations, etc.

Why CSS?

CSS saves Time - write once, use everywhere!

Pages load faster - less code means less bandwidth consumption & easy to maintain the content.

Easy Maintainance - can update.

Superior Styles to HTML - CSS provides more beautiful designs.

Multiple Device Compatibility - adjusts layout for mobiles, tablets, laptops.

Syntax- Selector { property : value ; } h1 { border : 1px; }

CSS Selectors :- selects the element that you want to style.

Element selector (tagname) - styles all element of specific type.

eg.
```
p { color : blue ;
    font-size : 18px ;
}
```
} applies to all <p> tags

ID selector (#idname) - styles a unique element with specific id.

eg.
```
#header { background-color : black ;
          color : white ;
          padding : 10px ;
}
```
} ID is unique! Only one element should have same id.

In html used as :- `<div id = "header" > Welcome ! </div>`

class Selector (.classname) - styles multiple elements with same class.

eg.
```
.button { background-color : green ;
          color : white ;
          padding : 10px ;
}
```
} Unlike id, multiple elements can share same class!

Universal selector (*) - styles to ALL elements on the page.

eg.
```
* { margin : 0 ;
    padding : 0 ;
}
```
} Makes everything have margin of 0

Grouping Selector (,) - styles multiple elements at once.

eg.
```
h1, h2, p { font-family : Arial , sans-serif ;
            color : darkgray ;
}
```
} Reduce repeated code & makes css cleaner.

Types of CSS

① Inline CSS - styles applied directly to HTML element using style attribute

Used when you need to style 1 element only.

eg. `<p style = "color : red" ; > Text </p>`

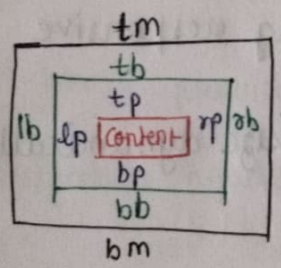
Pros - Quick & easy for small changes.

Cons - Not reusable hard to manage for large projects.

② Internal CSS - styles defined in <head> section within a <style> tag.
→ Used when you need to style a single ~~element in~~ HTML page.
→ eg. <head> <style> p { color : blue ; } </style> </ head>
→ Pros - keeps styling separate from content.
Affects the whole page without repeating styles.
→ Cons - Not reusable across multiple pages.
③ External CSS - styles in separate .csi file, linked via <link> tag.
→ Used when you need to style multiple pages in a website.
→ eg. <link rel = "stylesheet" href = "styles.css"> with p {color : green;} in styles.css
→ Pros - Best for large projects.
Keeps HTML clean & separate from styling.
→ Cons - Requires an extra file (.css) to be loaded.

Box Model in CSS :— All elements in DOM have a Box Model.
- The Box model of element is a box that contains the element.
- It is framework for how elements are rendered as retangular box, controlling sizes, spacing & layout.

Components -
1) Content - innermost part (text, images). Defined by 'width' & 'height' properties.
2) Padding - space inside the border, around content. Controlled by 'padding' properties.
3) Border - surrounds padding. Styled using 'border' property.
4) Margin - space outside the border. controlled by 'margin' property.



t → top
b → bottom
r → right
l → left

m → margin
b → border
p → padding

BOOTSTRAP
- It is a popular CSS framework that helps developers create responsive and stylish websites quickly.
- Most popular HTML, CSS & JS framework for development.

Why Bootstrap?
- Easy to use -
- Creates platform independent web-pages.
- Creates responsive web pages
- Has predefined components - buttons, cards, modals, etc.
- It is free and open source framework.

CSS over Bootstrap

Cascading Style Sheets

- Developed by Hakon Wium Lie, Bert Bos, WWW 17 Dec 1996.
- Used to design web pages.
- Does not have a grid system.
- Does not provide responsive pages or website.
- Here, we have to write code from scratch.
- More time-consuming for complex layouts.
- Best for unique, custom designs.

eg.
```
button { background-color: blue;
         color: white;
         padding: 10px 20px;
         border-radius: 5px;
}
```

- Best for designers who want full creative control.

Bootstrap

① Developed by Mark Otto, Jacob Thornton 19 Aug 2011.

② A CSS framework with prebuilt styles and components.

③ It is based on grid system.

④ We can design a responsive website or webpages.

⑤ We can add pre-defined class into the code without writing code.

⑥ Faster development with ready-to-use components.

⑦ Best for rapid development & consistency.

⑧ eg. `<button class="btn btn-primary">Click Me</button>`

⑨ Developers who need quick, responsive design.

Bootstrap Grid System

- It is a flexible layout system that helps in designing responsive web pages easily.
- It uses a 12 column layout, allowing elements to resize dynamically according to different screen sizes.
- It consist of 3 components:

Container :- These are outermost elements that hold the grid system.
`<div class="container">` ⟶ fixed width container.
`<div class="container-fluid">` → full width container.

Rows :- used to create horizontal group of columns. Each row should be placed inside a container & should contain one/more columns. `<div class="row">` → defines row

Columns :- this is where you place your content. (6/12 colm at medium screen)
`<div class="col-[breakpoint]-[size]">` eg. col-md-6

Breakpoint options - xs, sm, md, lg, xl, xxl, to adjust layout as per screen size.

eg:

```
<div class = "container">
  <div class = "row">
    <div class = "col-sm-4"> Column 1 </div>
    <div class = "col-sm-4"> Column 2 </div>
    <div class = "col-sm-4"> Column 3 </div>
  </div>
</div>
```

Bootstrap Components :—

Alerts - used to show success, error, warning or info messages.
- display alert messages.

eg.

```
<div class = "alert alert-success"> ... </div>
<div class = "alert alert-danger"> ... </div>
```

Buttons - used for actions like submitting forms, triggering events, etc.
- Bootstrap styles with 'btn' base class.
- available in different styles, sizes & with icons.

eg.

```
<button class = "btn btn-primary"> Primary </button>
<button class = "btn btn-success"> Success Link </button>
```

Cards - used to display grouped content like text, images, links or buttons in a boxed layout.

eg.

```
<div class = "card">
  <div class = "card-body">
    Hi, Hello !
  </div>
</div>
```

Navbar - provides a structured menu for navigation & adapts different screen sizes.
navigation headers for your website.

eg.

```
<nav class = "navbar">
  <a class = "navbar-brand" href = "#"> Navbar </a>
</nav>
```

Forms - allows users to enter text, emails, passwords, etc.
- collects info from user

eg.

```
<form>
  <input type = "text" class = "form-control" >
</form>
```

Dropdown - hidden menu that appears while clicking a button

Bootstrap as Cross Platform :-

It works seamlessly accross different devices, OS & browsers.
↳ Responsive Design ↳ Prebuilt Components ↳ offline usage.
↳ Browser compatibility ↳ Platform independence

W3C - World Wide Web Consortium

- It is an international organization that develops & maintains standards for WWW.
- founded by Tim Berners Lee in 1994.
- Develops guidelines, protocols & specifications for web technologies. Ensures compatibility & best practices across different browsers & devices.
- W3C is responsible for key web technologies like HTML, CSS & JS and others that are crucial for building & styling web pages & appln.

Advantages of W3C :-

1) Ensures web compatibility :- standardize web technologies.
2) Promotes accessibility :- more access ie to disabled ppl as well.
3) Enhance security :- strict guidelines.
4) Free & Open Standards :- everything is free & open
5) Supports latest web technologies :- supports AI, Web 3.0, AR/VR.

Disadvantages of W3C :-

1) Complexity for Begginners
2) Takes too long to approve new technologies.
3) No one is forced to follow it :- many companies just ignore the rules.
4) Browsers Don't always follow W3C.
5) Slow to catch up with new tech.

How W3C Supports Web Techologies ?

Supports by creating & maintaining open standards that ensures the web remains free, accessible & interoperability platform.

- Develops standards → creates rules for Web Tech. These ensure consistency across different browser & devices.
- Ensure Accessibility → includes WAI-ARIA & WCAG guidelines.
- Promotes security & Privacy
- Provides resources
- Encourage Innovations

Benefits :-

① Interoperability - website work consistently across devices & appln are developed
② Accessibility - usable by everyone.
③ Security - data is protected. ④ Innovation - new web tech.