

MILESTONE - 02

1) HTML - HYPER TEXT MARKUP LANGUAGE

- Used for creating and structuring web pages and web applications.
- Consists of series of elements and tags.
- `<!DOCTYPE html>` : Declaration of the HTML version.
- `<html>` : The root element of the HTML page.
- `<head>` : Contains meta-information about the HTML document, such as the title, character encoding(`<meta charset="UTF-8">`), and viewport (`<meta name="viewport" content="width=device-width, initial-scale=1.0">`) settings.
- `<meta name="viewport" content="width=device-width">` — This viewport element ensures the page renders at the width of viewport (helps in viewing the web page in both the desktop and mobile screen).
- `<title>` : Sets the title of the webpage, which appears in the browser's title bar or tab.
- `<body>` : Contains the visible content of the webpage.
- Various elements like `<header>` , `<nav>` , `<section>` , `<h1>` , `<h2>` , `<p>` , `` , `` , and `<footer>` : Used to structure and organize the content of the webpage.
- `<a>` : Creates hyperlinks to navigate within the website or to external pages.

2) XHTML - Extensible Hypertext Markup Language

- It is part of the XML (eXtensible Markup Language) family.
- It is essentially a stricter and more XML-compliant version of HTML. XHTML was designed to bridge the gap between HTML and XML

KEY FEATURES:

- Follows the syntax rules of XML, which means they must be properly structured, with all elements correctly nested and closed, and attribute values quoted.
- It allows developers to create custom elements and attributes using XML namespaces using the attribute “xmlns”.

- Unlike HTML, XHTML imposes stricter syntax requirements, such as lower-case tag names, attribute values enclosed in quotes, and self-closing tags for empty elements like `
` and ``.

3) HTML Elements and Tags:

- Html elements defines the various parts of a web page like headings, paragraphs, lists, links, images, forms, tables...as it conveys the search engine that how to display the content to the user.
- Elements are composed of tags and contents all together (`<p>Hello!!!</p>`).
- Tags :
 - `<html>`
 - `<head>`
 - `<title>`
 - `<meta>`
 - `<body>`
 - `<h1....h6>`
 - `<p>`
 - `<a>`
 - ``
 - ` → <dl>`
 - `<div>`
 - `` etc....

3) HTML Attributes :

- They are defined within the opening tag of an element, consists of a name and value separated by "=", enclosed in ("") or (' ').
- Modifies the behavior or appearance of elements.
- Attributes :

- *id*
- *class*
- *style*
- *src*
- *href*
- *alt*
- *disabled*
- *required.....*

4) HTML Class :

- Used to specify and apply CSS to ***one or more*** class names for an HTML element.
- Can contain letters, digits, hyphens, and underscores, but cannot start with a digit.
- “***.classname***” is used in style.
- can be applied multiple times.

5) HTML id :

- The `id` attribute is used to specify a unique identifier for an HTML element.
- The `id` should only be used once per HTML document, as each identifier must be unique within the document.
- `#id_Name` is used in style.

6) Formatting :

1. Text
2. Lists
3. Alignment
4. Line breaks & Horizontal Rules (br & hr)

5. Quotations
6. Attributes
7. Preformatted text
8. Special characters

7) Charsets :

- UTF - Unicode Transformation Format
- Used for character encoding for Web Content.

8) Semantic Elements :

Semantic elements are those that convey meaning about the content they contain, making it more accessible to both users and search engines. (eg., `header` , `nav` , `main` , `section` , `article` , `footer` .)

9) Non-Semantic :

Nonsemantic elements do not convey any inherent meaning about the content they contain. They are typically used for layout and styling purposes. (eg., `div` , `span` , `br` , `hr` , `i` , `b` , `u` .

10) DOM - Document Object Model. (Represents web page in a structured format)

- A programming interface for web documents.
- The DOM represents the structure of a document as a **tree** where each node represents a part of the document, such as elements, attributes, and text.
- Web browsers use the DOM to render web pages and to allow scripting languages, such as JavaScript, to interact with the content of the page dynamically.
- JavaScript cannot understand HTML directly, So it interprets and interacts with DOM which is created by the browser.
- DOM is not a part of JavaScript, It is a part of WEB API's and it has an ability to interact with JavaScript.

- DOM Manipulation :
 - Creating
 - Modifying
 - Adding Event Listeners
 - Styling, Removing, Cloning, Replacing
 - Traversing
- DOM methods :
 - getElementById()
 - getElementsByClassName()
 - getElementsByTagName()
 - querySelector()
 - querySelectorAll()
 - createElement(tagName)
 - appendChild(node), removeChild(node)
 - setAttribute(name,value)
 - addEventListener(event,function)
- DOM Properties:
 - innerHTML
 - textContent
 - value
 - className
 - style