Level 1

HTML

is the standard markup language for creating web pages and applications. It stands for HyperText Markup Language and is used to structure content and define the layout of a webpage.

First HTML File : index.html

To create a basic HTML file, open a text editor and save a blank file with the .html extension. This file will serve as the starting point for your webpage. It is convention to name the first HTML file as index.html, as many web servers will default to this file when a folder is accessed without specifying a specific file.

HTML Tag

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

HTML tags are used to define different elements within a webpage, such as paragraphs, headings, images, and links. Each tag has a specific purpose, and can be modified with attributes to control the way they appear on the page. The example tag is used to define a paragraph of text.

Basic HTML Page

A basic HTML page structure includes the `<!DOCTYPE html>` declaration, which tells the browser that the page is an HTML5 document. The `<html>` tag defines the root of the document, and within it, the `<head>` and `<body>` tags contain information about the document and its contents. These tags are essential for creating any HTML page.

```html
<html>
<head>
  <title>My First Webpage</title>
</head>
<body>
  <h1>Welcome to My First Webpage</h1>
  <p>This is a basic example of an HTML page.</p>
</body>
</html>
```
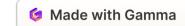
Comments in HTML

HTML, comments are used to add notes or explanations to the code that are not visible on the webpage. Comments in HTML are written using the `<!-- -->` syntax.

```html
<!-- This is an example of a comment in HTML -->
```

HTML is NOT case sensitive

. This means that you can write HTML tags and attributes in either uppercase or lowercase letters. For example, `<h1>` and `<H1>` would produce the same heading element. However, it is common practice to write HTML in lowercase letters to improve code readability and consistency.

Level 2

Basic HTML Tags

HTML tags are the building blocks of a webpage. Here are some basic HTML tags that are commonly used:

Document Structure Tags:

1. `<html>`: Defines the root of an HTML document.
2. `<head>`: Contains metadata and links to stylesheets or scripts.
3. `<body>`: Encloses the content of the webpage.
4. `<title>`: Sets the title of the webpage.
5. `<meta>`: Provides metadata like character encoding and viewport settings.
6. `<link>`: Links external resources like stylesheets.
7. `<style>`: Adds internal CSS styles.
8. `<script>`: Embeds or references JavaScript.
9. `<base>`: Sets the base URL for relative links.

Text and Typography Tags:

1. `<p>`: Defines a paragraph.
2. `<h1>` **to** `<h6>`: Represent headings, from largest to smallest.
3. `<span>`: Used for styling or manipulating specific text inline.
4. `<strong>`: Indicates strong importance (bold by default).
5. `<em>`: Emphasizes text (italicized by default).
6. `<b>`: Makes text bold without semantic meaning.
7. `<i>`: Makes text italic without semantic emphasis.
8. `<u>`: Underlines text.
9. `<small>`: Displays smaller, less significant text.
10. `<mark>`: Highlights text to indicate relevance.
11. `<sub>`: Displays subscript text.
12. `<sup>`: Displays superscript text.
13. `<ins>`: Represents inserted text (usually underlined).
14. `<del>`: Represents deleted text (usually strikethrough).
15. `<br>`: Inserts a line break.
16. `<wbr>`: Suggests a break opportunity in long text or URLs.

Links and Navigation Tags:

1. `<a>`: Creates hyperlinks to navigate between pages.
2. `<nav>`: Represents a section with navigation links.

List Tags:

1. `<ul>`: Defines an unordered list with bullet points.
2. `<ol>`: Defines an ordered list with numbers or letters.
3. `<li>`: Represents an item in a list.
4. `<dl>`: Defines a description list.
5. `<dt>`: Represents a term in a description list.
6. `<dd>`: Represents the description definition.

Multimedia Tags:

1. `<img>`: Embeds images.
2. `<audio>`: Embeds audio with playback controls.
3. `<video>`: Embeds videos with playback controls.
4. `<source>`: Specifies media resources for `<audio>` or `<video>`.
5. `<figure>`: Groups media content like images with captions.
6. `<figcaption>`: Provides a caption for a `<figure>` element.

Table Tags:

1. `<table>`: Displays data in a tabular format.
2. `<tr>`: Defines a table row.
3. `<td>`: Represents a cell in a table.
4. `<th>`: Represents a header cell in a table.
5. `<thead>`: Groups the header rows in a table.
6. `<tbody>`: Groups the body rows in a table.
7. `<tfoot>`: Groups the footer rows in a table.

Form Tags:

1. `<form>`: Creates a form for data submission.
2. `<input>`: Collects user input (text, password, checkbox, etc.).
3. `<textarea>`: Defines a multi-line text input.
4. `<button>`: Creates a clickable button.
5. `<label>`: Labels form elements for accessibility.
6. `<fieldset>`: Groups related form elements.
7. `<legend>`: Provides a title for a `<fieldset>`.
8. `<select>`: Creates a dropdown list.
9. `<option>`: Defines an option in a dropdown list.
10. `<optgroup>`: Groups options in a dropdown list.
11. `<datalist>`: Provides predefined options for an `<input>` field.

Semantic Structure Tags:

1. `<header>`: Represents the header section of a document.
2. `<footer>`: Represents the footer section of a document.
3. `<main>`: Represents the main content of a document.
4. `<section>`: Defines a thematic grouping of content.
5. `<article>`: Represents self-contained content (e.g., blog post).
6. `<aside>`: Represents side content, like sidebars or callouts.
7. `<address>`: Represents contact information.

Programmatic and Metadata Tags:

1. `<code>`: Represents inline programming code.
2. `<pre>`: Displays preformatted text.
3. `<kbd>`: Represents keyboard input.
4. `<var>`: Represents a variable or mathematical expressions ($A=\pi r2$).
5. `<details>`: Creates a collapsible section.
6. `<summary>`: Provides a summary or label for `<details>`.

These are just a few examples of the many HTML tags available for different purposes.

Attributes provide additional information or modify the behavior of HTML tags. They are specified within the opening tag of an HTML element and are written as name-value pairs. For example, the `href` attribute is used with the `<a>` tag to specify the URL of the hyperlink. Other commonly used attributes include `src` for specifying the source of an image and `class` for applying CSS styles to specific elements. Attributes are an important part of HTML and greatly enhance the functionality and appearance of webpages.