

HTML EMG

HTML (HyperText Markup Language) is the standard markup language for creating web pages. It describes the structure of a web page using a series of elements.

This Module is consist of 6 parts

```
bali-king@war-machine:~/BaliGit/KintsugiStack/HTML_EMG$ tree
```

```
.
├── HTML
│   ├── images
│   │   ├── Screenshot from 2024-05-31 19-37-23.png
│   │   └── Screenshot from 2024-05-31 19-37-59.png
│   ├── index.html # P6
│   └── pages
│       ├── Compiled.htm # COMPILED
│       ├── p1.html # P1
│       ├── p2.html # P2
│       ├── p3.html # P3
│       ├── p4.html # P4
│       └── p5.html # P5
└── README.md
```

4 directories, 10 files

- Download [LiveServer](#) Extension in VSC, easy to setup Dev.Server GUI

Table of contents

- [HTML EMG](#)
 - [Table of contents](#)
- [Part 1: HTML Structure and Basic Elements](#)
 - [1. Introduction to HTML Document Structure](#)
 - [1.1 Basic HTML Structure](#)
 - [2. Navigation and Links](#)
 - [2.1 Navigation Example](#)
 - [2.2 Link Tags](#)
 - [3. HTML Comments](#)
 - [3.1 Example of a Comment](#)
 - [3.2 Multiline Comment](#)
- [Part 2: Text Formatting and Multimedia Elements](#)
 - [1. HTML Elements for Text Formatting](#)
 - [1.1 Headings](#)
 - [1.2 Paragraphs](#)
 - [1.3 Line Breaks](#)
 - [1.4 Horizontal Rule](#)

- 1.5 Text Formatting Tags
 - 1.6 Superscript and Subscript
 - 1.7 Special Characters
- 2. Multimedia Elements
 - 2.1 Images
 - 2.2 Videos
 - 2.3 Links with Images
- 3. Preformatted Text
- Part 3: Understanding How Websites Work and Client-Side Editing
 - 1. How Websites Are Delivered to Clients
 - 1.1 Request and Response
 - 1.2 Basic Flow of How a Web Page is Delivered
 - 2. Accessing and Editing Websites in the Browser Console
 - 2.1 Opening Developer Tools
 - 2.2 Editing HTML in the Browser
 - 2.3 Editing CSS
 - 2.4 Running JavaScript in the Console
 - 3. Downloading Web Pages
 - 3.1 Saving a Webpage
 - 3.2 Using the Console to Download Resources
 - 4. Example: Downloading LinkedIn's Page
- Part 4: Advanced HTML Elements and Semantic Structure
 - 1. Advanced Structure Tags
 - 1.1 Header and Footer
 - 1.2 Navigation (`<nav>`)
 - 1.3 Section and Article
 - 1.4 Div and Span
 - 2. Additional HTML Elements
 - 2.1 Aside
 - 2.2 Blockquote and Quotation
 - 2.3 Address, Cite, Code, and More
 - 2.4 Preformatted Text
 - 3. Images and Captions
- Part 5: Forms, Lists, and Embedding External Content
 - 1. Lists in HTML
 - 1.1 Ordered List
 - 1.2 Unordered List
 - 1.3 Definition List
 - 2. Tables
 - 3. Forms in HTML
 - 3.1 Basic Form Structure
 - 3.2 Form Elements
 - 4. Embedding External Content
 - 4.1 Embedding a YouTube Video
- Final Part: Homepage and Basic Link Navigation
 - 1. Homepage Layout

- [1.1 Simple Homepage Structure](#)
- [1.2 Link Structure](#)

Part 1: HTML Structure and Basic Elements

```
<!DOCTYPE html>
<html>
<head><title>Boiler Plateeee...</title></head>
  <body>
    <nav>
      <ul>
        <li><a href="../index.html">Back to Homepage</a></li>
        <li><a href="p1.html">Page 1</a></li>
        <li><a href="p2.html">Page 2</a></li>
        <li><a href="p3.html">Page 3</a></li>
        <li><a href="p4.html">Page 4</a></li>
        <li><a href="p5.html">Page 5</a></li>
      </ul>
    </nav>
    <p>Boilerplate</p>

    <!-- comment
    is
    multiline-->
  </body>
</html>
```

- save with file extension `.htm`, `.html`

1. Introduction to HTML Document Structure

Every HTML document follows a basic structure, including the `DOCTYPE`, `<html>`, `<head>`, and `<body>` elements. Let's break down a simple HTML document:

1.1 Basic HTML Structure

```
<!DOCTYPE html>
<html>
<head>
  <title>Boiler Plateeee...</title>
</head>
<body>
  <!-- Content goes here -->
</body>
</html>
```

- `<!DOCTYPE html>`: This declaration defines the document type as HTML5.

- `<html>`: The root element that contains all HTML content.
- `<head>`: Contains metadata like the document title (`<title>`), which appears in the browser tab.
- `<body>`: Contains all the content that will be visible on the web page.

2. Navigation and Links

HTML provides the `<nav>` element for marking navigation sections in the document. Inside it, we use the `` (unordered list) and `` (list item) elements to structure navigation links.

2.1 Navigation Example

```
<nav>
  <ul>
    <li><a href="../index.html">Back to Homepage</a></li>
    <li><a href="p1.html">Page 1</a></li>
    <li><a href="p2.html">Page 2</a></li>
    <li><a href="p3.html">Page 3</a></li>
    <li><a href="p4.html">Page 4</a></li>
    <li><a href="p5.html">Page 5</a></li>
  </ul>
</nav>
```

- `<nav>`: Defines the navigation section of the page.
- ``: Represents an unordered list of items.
- ``: Each list item contains a link represented by the `<a>` tag.
- `href`: Specifies the destination URL of the link.

2.2 Link Tags

The `<a>` tag is used to create links. The `href` attribute is required to define the link destination.

```
<a href="p1.html">Page 1</a>
```

- This creates a link to `p1.html`.

3. HTML Comments

In HTML, comments are written between `<!--` and `-->`. These comments are not displayed in the browser but are visible in the source code. Comments are helpful for developers to explain code or mark sections for later modification.

3.1 Example of a Comment

```
<!-- This is a comment -->
```

3.2 Multiline Comment

```
<!--
  comment
  is
  multiline
-->
```

- Comments can span multiple lines, as shown in the example above.

Part 2: Text Formatting and Multimedia Elements

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <nav>
    <ul>
      <li><a href="../index.html">Back to Homepage</a></li>
      <li><a href="p1.html">Page 1</a></li>
      <li><a href="p2.html">Page 2</a></li>
      <li><a href="p3.html">Page 3</a></li>
      <li><a href="p4.html">Page 4</a></li>
      <li><a href="p5.html">Page 5</a></li>
    </ul>
  </nav>
  <div id="id1">
    <h1>Heading 1</h1>
    <h2>Heading 2</h2>
    <h3>Heading 3</h3>
    <h4>Heading 4</h4>
    <h5>Heading 5</h5>
    <h6>Heading 6</h6>
  </div>
  <div id="id2">
    <p>Paragraph 1</p>
    <p>Paragraph 2 automactic spacing ,TEXT WRAP :0</p>
    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Et illo
<BR> ipsum autem dolore quis aut iusto <br>sit dolorem obcaecati in!

    </br>
    <hr>
    
```

```

        <video src="https://www.w3schools.com/html/mov_bbb.mp4"
controls width="360" autoplay muted></video>
        <a href="https://www.google.com">Google</a>
        <a href="https://www.google.com" target="_blank">[at target="_blank"]</a>
        <b>Bold</b>
        <i>Italic</i>
        <u>Underline</u>
        <s>Strikethrough</s>

    </p>

</div>

<div id="id3">
    <pre>
        Preformatted text
            Line 1
        Line 2
    Line 3
    Line 4
Line 5

    <big>big</big>
    <small>small</small>
    H<sub>2</sub>
    n<sup>2</sup>

    &copy;Kintsugi-Programmer

    e=mc<sup>2</sup>

    </pre>
</div>

</body>
</html>

```

1. HTML Elements for Text Formatting

HTML provides a variety of elements to format text and structure content. These elements are commonly used to emphasize text, create headings, and display special characters.

1.1 Headings

Headings are used to structure content, and HTML offers six levels of headings, from `<h1>` (most important) to `<h6>` (least important).

```
<h1>Heading 1</h1>
<h2>Heading 2</h2>
<h3>Heading 3</h3>
<h4>Heading 4</h4>
<h5>Heading 5</h5>
<h6>Heading 6</h6>
```

1.2 Paragraphs

The `<p>` element defines a paragraph. Paragraphs automatically include some space above and below, which is useful for separating text content.

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

1.3 Line Breaks

The `
` element is used to insert a line break within content, ensuring text is displayed on a new line.

```
<p>Lorem ipsum dolor sit amet.</p>
<br>
<p>Text on the next line.</p>
```

1.4 Horizontal Rule

The `<hr>` tag is used to create a horizontal line, which is useful for dividing sections of a page.

```
<hr>
```

1.5 Text Formatting Tags

These tags modify the appearance of text for emphasis or styling.

- **Bold:** ``
- **Italic:** `<i>`
- **Underline:** `<u>`
- **Strikethrough:** `<s>`
- **Big:** `<big>`
- **Small:** `<small>`

```
<b>Bold text</b>
<i>Italic text</i>
<u>Underlined text</u>
<s>Strikethrough text</s>
```

```
<big>Larger text</big>  
<small>Smaller text</small>
```

1.6 Superscript and Subscript

HTML allows the use of superscript (`<sup>`) and subscript (`<sub>`) tags for mathematical or scientific notation.

```
H<sub>2</sub>O  
x<sup>2</sup>
```

1.7 Special Characters

Special characters, such as the copyright symbol, can be inserted using character entities.

```
&copy;Kintsugi-Programmer
```

2. Multimedia Elements

HTML allows the embedding of various types of multimedia, such as images and videos, to enhance user experience.

2.1 Images

Images are added using the `` tag. The `src` attribute defines the image source, and `alt` provides alternative text for screen readers. You can also define width and height.

```

```

2.2 Videos

HTML supports embedding videos using the `<video>` element. The `controls` attribute adds basic video controls like play, pause, and volume. You can also set the `autoplay` and `muted` attributes.

```
<video src="https://www.w3schools.com/html/mov_bbb.mp4" controls  
width="360" autoplay muted></video>
```

2.3 Links with Images

Links can also contain images, and you can define the `target="_blank"` attribute to open the link in a new tab.

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

3. Preformatted Text

The `<pre>` tag preserves both spaces and line breaks, which is useful for displaying code or structured text that requires formatting.

```
<pre>
  Preformatted text
    Line 1
  Line 2
  Line 3
  Line 4
  Line 5
</pre>
```

Part 3: Understanding How Websites Work and Client-Side Editing

1. How Websites Are Delivered to Clients

When you visit a website, the server sends different types of files—HTML, CSS, and JavaScript—that the browser processes to display the webpage. Here's an overview of the process:

1.1 Request and Response

1. **User Request:** When you type a website URL (e.g., `www.linkedin.com`) into the browser's address bar, your browser sends a request to the server that hosts the website. This request is typically done via HTTP or HTTPS.
2. **Server Response:** The server processes the request and sends back the HTML, CSS, and JavaScript files to your browser.
3. **Rendering:** The browser then renders these files, combining HTML for structure, CSS for styling, and JavaScript for interactivity, to display the website as you see it on your screen.

1.2 Basic Flow of How a Web Page is Delivered

- **HTML:** Defines the structure of the webpage (e.g., headings, paragraphs, links).
- **CSS:** Provides styles for the elements in HTML (e.g., colors, layout).
- **JavaScript:** Adds interactivity, like animations, form validation, or dynamic content updates.

For example, if you visit LinkedIn's homepage, the following files might be involved:

- An HTML file that defines the page structure.
- CSS files that define the layout and design of the page.
- JavaScript files that load dynamic content and enable interactivity.

2. Accessing and Editing Websites in the Browser Console

You can use the browser's developer tools to interact with any website. These tools allow you to inspect, edit, and download various elements of the page.

2.1 Opening Developer Tools

To open the developer tools in most browsers, you can:

- Right-click anywhere on the page and select **Inspect** (Chrome, Firefox, Edge).
- Press **F12** or **Ctrl+Shift+I** (Windows) / **Cmd+Opt+I** (Mac).

The developer tools are divided into several sections, including:

- **Elements:** Allows you to view and modify the HTML and CSS.
- **Console:** Lets you run JavaScript commands directly in the browser.
- **Network:** Shows all the network requests (like images, CSS, JS files) made by the webpage.

2.2 Editing HTML in the Browser

- **Inspect the HTML:** In the "Elements" tab, you can navigate the HTML structure. Right-click any element and click **Edit as HTML** to make changes directly on the page.

```
<div class="profile-header">
  <h1>John Doe</h1>
  <p>Web Developer</p>
</div>
```

You can modify text, add new elements, or change the structure directly in the console. This change will only be reflected locally in your browser and will not affect the live website.

2.3 Editing CSS

- **Modify Styles:** In the "Styles" section of the developer tools, you can edit the CSS properties of any element. For example:

```
.profile-header {
  background-color: #000;
```

```
    color: #fff;  
}
```

You can adjust the font size, color, padding, and many other styles in real time.

2.4 Running JavaScript in the Console

- **Console Command:** You can use the console to run JavaScript commands. For example, to change the text content of an element:

```
document.querySelector('.profile-header h1').textContent = 'Jane Doe';
```

This JavaScript code will change the name displayed in the `h1` tag.

3. Downloading Web Pages

You can also download the entire HTML, CSS, and JS files of a webpage for offline use or to analyze the website's structure.

3.1 Saving a Webpage

In most browsers:

1. Right-click on the page and select **Save As**.
2. Choose **Webpage, Complete** to download all the assets (HTML, CSS, images, etc.).

Alternatively, you can use the **Network tab** in the developer tools to monitor network requests and download individual resources, such as images or scripts.

3.2 Using the Console to Download Resources

You can download specific files directly through the browser's console by using JavaScript. For example, to download an image:

```
var link = document.createElement('a');  
link.href = 'https://www.example.com/image.jpg';  
link.download = 'image.jpg';  
link.click();
```

This will trigger the download of the image to your local system.

4. Example: Downloading LinkedIn's Page

Let's walk through how you can inspect and download LinkedIn's homepage:

1. **Inspect the Page:**

- Open LinkedIn in your browser.
- Right-click and select **Inspect** to open the developer tools.
- Browse through the HTML structure and observe how LinkedIn is structured using HTML, CSS, and JavaScript.

2. Save the HTML:

- Go to the **Elements** tab and right-click on the `<html>` element to copy the outer HTML.
- Paste this into a new `.html` file on your system.

3. Download LinkedIn Resources:

- In the **Network** tab, you can see all the network requests for assets like images, JavaScript files, and stylesheets.
- Right-click and select **Open in new tab** to save images or files.

4. Download JavaScript and CSS:

- Use the "Sources" or "Network" tabs to view and download JavaScript and CSS files linked to the page.

Part 4: Advanced HTML Elements and Semantic Structure

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <header><big>Header</big></header>
  <hr>
  <nav>
    <ul>
      <li><a href="../index.html">Back to Homepage</a></li>
      <li><a href="p1.html">Page 1</a></li>
      <li><a href="p2.html">Page 2</a></li>
      <li><a href="p3.html">Page 3</a></li>
      <li><a href="p4.html">Page 4</a></li>
      <li><a href="p5.html">Page 5</a></li>
    </ul>
  </nav>
  <hr>
  <nav>Nav<br>
    Navigation Tags<br>
    <a href="#Article 1">Article 1</a><br>
    <a href="#Article 2">Article 2</a><br>
    <a href="#Article 3">Article 3</a><br>
```

```
</nav>
<hr>
<main>
  <section id="Article 1">
    <article>
      <header>Article1</header>
      <p>Lorem ipsum dolor sit<div>[<small>Div tag used</small>]
</div>amet, consectetur adipisicing elit. Itaque repellendus aut quasi
dolore reiciendis nostrum totam explicabo aliquam animi a ipsa atque
expedita enim molestiae mollitia adipisci numquam officia quisquam velit,
perspiciatis aperiam! Distinctio tempore, veniam incidunt alias labore
quasi at consequatur commodi soluta odio fuga, eligendi dolor placeat.</p>
    </article>
  </section>

  <section id="Article 2">
    <article>
      <header>Article2</header>
      <p>Lorem ipsum dolor sit<span>[<small>Span tag used</small>]
</span>amet, consectetur adipisicing elit. Itaque repellendus aut quasi
dolore reiciendis nostrum totam explicabo aliquam animi a ipsa atque
expedita enim molestiae mollitia adipisci numquam officia quisquam velit,
perspiciatis aperiam! Distinctio tempore, veniam incidunt alias labore
quasi at consequatur commodi soluta odio fuga, eligendi dolor placeat.</p>
    </article>
  </section>

  <section id="Article 3">
    <article>
      <header>Article3</header>
      <p>Lorem ipsum dolor sitamet, consectetur adipisicing elit.
Itaque repellendus aut quasi dolore reiciendis nostrum totam explicabo
aliquam animi a ipsa atque expedita enim molestiae mollitia adipisci
numquam officia quisquam velit, perspiciatis aperiam! Distinctio tempore,
veniam incidunt alias labore quasi at consequatur commodi soluta odio fuga,
eligendi dolor placeat.</p>
    </article>
  </section>
  <div>
    Div <br>
    <br><caption>Image Caption</caption><br>
    
    <br><caption>Image Caption</caption><br>
  </div>

  <aside>
    <h3>Aside</h3>
    <q>Quotation</q>
    <blockquote>Blockquote</blockquote>
    <address>Address</address>
    <cite>Cite</cite>
    <code>Code</code>
```

```
<kbd>Keyboard</kbd>
<samp>Sample</samp>
<var>Variable</var>
<pre>Preformatted text
Typical URL Due to #{Navigation Tag}:
http://127.0.0.1:5500/Parabellum/Html/p4/index.html#Article%202
[SEE THIS
ABOVE #Article] </pre>
</aside>
</main>
<hr>
<footer>Footer</footer>

</body>
</html>
```

1. Advanced Structure Tags

HTML provides advanced tags that enhance the semantic structure of a page. These tags not only improve accessibility but also help with search engine optimization (SEO) by giving meaning to the content.

1.1 Header and Footer

- **<header>**: Defines the header for a document or a section.
- **<footer>**: Defines the footer for a document or a section. It usually contains information like copyright, contact links, or additional navigation.

```
<header><big>Header</big></header>
<hr>
```

```
<footer>Footer</footer>
```

1.2 Navigation (<nav>)

The **<nav>** element defines a section of links for navigation. It is typically used for menus, pagination, or other types of link groups.

```
<nav>
  <ul>
    <li><a href="../index.html">Back to Homepage</a></li>
    <li><a href="p1.html">Page 1</a></li>
    <li><a href="p2.html">Page 2</a></li>
    <li><a href="p3.html">Page 3</a></li>
    <li><a href="p4.html">Page 4</a></li>
    <li><a href="p5.html">Page 5</a></li>
```

```
</ul>
</nav>
```

1.3 Section and Article

- **<section>**: Defines a section of a webpage, typically with a heading. Sections are thematic grouping elements that allow for better content organization.
- **<article>**: Defines a self-contained, independent piece of content.

```
<section id="Article 1">
  <article>
    <header>Article1</header>
    <p>Lorem ipsum dolor sit amet...</p>
  </article>
</section>
```

1.4 Div and Span

- **<div>**: A block-level element used for grouping content together and applying styles.
- ****: An inline element used to apply styles or classes to a small portion of text or an inline element.

```
<div>
  
  <caption>Image Caption</caption>
</div>

<span>Some text here with styling</span>
```

2. Additional HTML Elements

2.1 Aside

The **<aside>** element is used for content that is tangentially related to the content around it, such as sidebars, advertisements, or additional information.

```
<aside>
  <h3>Aside</h3>
  <q>Quotation</q>
  <blockquote>Blockquote</blockquote>
  <address>Address</address>
  <cite>Cite</cite>
  <code>Code</code>
  <kbd>Keyboard</kbd>
  <samp>Sample</samp>
  <var>Variable</var>
```

```
<pre>Preformatted text</pre>
</aside>
```

2.2 Blockquote and Quotation

- **<q>**: Defines a short inline quotation.
- **<blockquote>**: Used to indicate longer quotes or citations from other sources.

```
<q>Quotation</q>
<blockquote>Blockquote</blockquote>
```

2.3 Address, Cite, Code, and More

- **<address>**: Represents contact information for a person or organization.
- **<cite>**: Used to reference the title of a work (e.g., a book, article, or webpage).
- **<code>**: Represents a fragment of computer code.

```
<address>123, Elm Street, NY</address>
<cite>Shakespeare's Hamlet</cite>
<code>console.log('Hello, World!');</code>
```

2.4 Preformatted Text

- **<pre>**: Used to display preformatted text, where whitespace and newlines are preserved. It is ideal for displaying code or structured text.

```
<pre>Preformatted text
Typical URL Due to Navigation Tag:
http://127.0.0.1:5500/Parabellum/Html/p4/index.html#Article%202
</pre>
```

3. Images and Captions

The **** tag is used to embed images into a webpage. The **caption** element is used to provide a description of the image, typically placed within a **<figure>** tag.

```

<caption>Image Caption</caption>
```

Part 5: Forms, Lists, and Embedding External Content

1. Lists in HTML

HTML provides three types of lists: **Ordered Lists (ol)**, **Unordered Lists (ul)**, and **Definition Lists (dl)**. These lists are useful for presenting items or definitions in a structured way.

1.1 Ordered List

The `` element creates an ordered (numbered) list. The `type` attribute specifies the numbering format:

- `1` for numbers (default),
- `a` for lowercase letters,
- `A` for uppercase letters,
- `i` for lowercase Roman numerals,
- `I` for uppercase Roman numerals.

```
<ol type="1">
  <li>Ordered List Item 1</li>
  <ol type="a">
    <li>Ordered List Item 1.1</li>
    <ol type="i">
      <li>Ordered List Item 1.1.1</li>
      <ol type="A">
        <li>Ordered List Item</li>
      </ol>
    </ol>
    <li>Ordered List Item 1.1.2</li>
  </ol>
  <li>Ordered List Item 2</li>
  <li>Ordered List Item 3</li>
  <li>Ordered List Item 4</li>
</ol>
```

1.2 Unordered List

The `` element creates an unordered (bulleted) list.

```
<ul>
  <li>Unordered List Item 1</li>
  <li>Unordered List Item 2</li>
  <li>Unordered List Item 3</li>
</ul>
```

1.3 Definition List

The `<dl>` element defines a definition list, with `<dt>` for terms and `<dd>` for definitions.

```
<dl>
  <dt>Definition Term 1</dt>
  <dd>Definition Description 1</dd>
  <dt>Definition Term 2</dt>
  <dd>Definition Description 2</dd>
</dl>
```

2. Tables

HTML allows you to create tables to organize data into rows and columns. The `<table>` element is used to define a table, with `<tr>` for rows, `<th>` for headers, and `<td>` for table data.

```
<table border="1">
  <caption>Table Caption</caption>
  <tr>
    <th>Table Header 1</th>
    <th colspan="2">Table Header 2</th>
  </tr>
  <tr>
    <td>Table Data 1</td>
    <td>Table Data 2</td>
    <td>Table Data 3</td>
  </tr>
  <tr>
    <td>Table Data 4</td>
    <td>Table Data 5</td>
    <td>Table Data 6</td>
  </tr>
</table>
```

3. Forms in HTML

Forms are used to collect user input. The `<form>` element contains various input fields such as text boxes, checkboxes, radio buttons, dropdowns, and submit buttons.

3.1 Basic Form Structure

```
<form action="submit.php" method="post">
  <label for="name">Name:</label>
  <input type="text" name="name" id="name" placeholder="Name">
  <br>
  <label for="email">Email:</label>
  <input type="email" name="email" id="email" placeholder="Email">
  <br>
  <input type="submit" value="Submit">
</form>
```

3.2 Form Elements

- **Text Input:** `<input type="text">`
- **Password Input:** `<input type="password">`
- **Email Input:** `<input type="email">`
- **Select Dropdown:** `<select>` and `<option>`
- **Checkbox:** `<input type="checkbox">`
- **Radio Button:** `<input type="radio">`
- **Textarea:** `<textarea>`
- **Button:** `<button>`

Example of a form with various inputs:

```
<form>
  <label for="name">Name:</label>
  <input type="text" id="name" placeholder="Your name">
  <br>
  <label for="email">Email:</label>
  <input type="email" id="email" placeholder="Your email">
  <br>
  <label for="branch">Branch:</label>
  <select id="branch">
    <option value="CSE">Computer Science</option>
    <option value="ECE">Electronics</option>
  </select>
  <br>
  <label for="date">Date:</label>
  <input type="date" id="date">
  <br>
  <label for="color">Color:</label>
  <input type="color" id="color">
  <br>
  <label for="range">Range:</label>
  <input type="range" id="range">
  <br>
  <input type="submit" value="Submit">
</form>
```

4. Embedding External Content

HTML allows you to embed external content such as videos, maps, and other web pages within your website.

4.1 Embedding a YouTube Video

You can embed a YouTube video using the `<iframe>` element. The `src` attribute contains the URL of the video.

```
<iframe src="https://www.youtube.com/embed/3JZ_D3ELw0Q" width="560"
height="315" frameborder="0" allowfullscreen></iframe>
```

Final Part: Homepage and Basic Link Navigation

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  Homepage
  <ul>
    <li><a href="pages/p1.html">Page 1</a></li>
    <li><a href="pages/p2.html">Page 2</a></li>
    <li><a href="pages/p3.html">Page 3</a></li>
    <li><a href="pages/p4.html">Page 4</a></li>
    <li><a href="pages/p5.html">Page 5</a></li>
  </ul>

</body>
</html>
```

1. Homepage Layout

In HTML, you can easily create a homepage that links to other pages within the website. The homepage often serves as the starting point or the main index page, providing links to other important sections.

1.1 Simple Homepage Structure

Here's an example of a basic homepage layout with links to other pages:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  Homepage
  <ul>
    <li><a href="pages/p1.html">Page 1</a></li>
    <li><a href="pages/p2.html">Page 2</a></li>
    <li><a href="pages/p3.html">Page 3</a></li>
    <li><a href="pages/p4.html">Page 4</a></li>
    <li><a href="pages/p5.html">Page 5</a></li>
  </ul>
```

```
</body>  
</html>
```

- The `` (unordered list) is used to group the navigation links together.
- Each `` (list item) contains an `<a>` (anchor) tag that links to different pages (`p1.html`, `p2.html`, etc.) inside the `pages` folder.

1.2 Link Structure

The links (``) point to pages located within the "pages" directory. This makes it easy to organize your website into multiple sections.

```
<li><a href="pages/p1.html">Page 1</a></li>
```

- `href="pages/p1.html"`: Points to the file `p1.html` located in the `pages` folder.