

# Module 2 ( Mernstack HTML)

## • HTML Basics

### Theory Assignment:

**Question 1: Define HTML. What is the purpose of HTML in web development?**

HyperText Markup Language is the standard language used to create and structure content on the web. It defines the elements and structure of a web page using a system of tags, attributes, and content.

### Purpose:

The purpose of HTML in web development is to structure and organize web content using a system of tags and elements. It defines the layout of a webpage, enabling the integration of text, images, links, and multimedia.

**Question 2: Explain the basic structure of an HTML document. Identify the mandatory tags and their purposes?**

The basic structure of an HTML document includes the following mandatory tags:

1. `<!DOCTYPE html>`: Declares the document type and version of HTML being used.
2. `<html>`: The root element that encompasses the entire HTML document.
3. `<head>`: Contains metadata, such as the document title, character encoding, and links to stylesheets or scripts.
4. `<title>`: Specifies the title of the document, displayed on the browser tab.
5. `<body>`: Contains the visible content of the webpage, including text, images, links, and other elements.

**Question 3: What is the difference between block-level elements and inline elements in HTML?**

**Provide examples of each.**

- **Block-Level Elements:** Start on a new line and take up the full width of their container. Used for structuring larger sections of a webpage.  
**Examples:** `<div>`, `<p>`, `<h1>`, `<ul>`, `<table>`.
- **Inline Elements:** Do not start on a new line and only take up as much width as their content requires. Used within block-level elements for styling or grouping.  
**Examples:** `<span>`, `<a>`, `<img>`, `<strong>`, `<em>`.

**Question 4: Discuss the role of semantic HTML. Why is it important for accessibility and SEO?**

**Provide examples of semantic elements?**

**Role of Semantic HTML:**

Semantic HTML refers to the use of HTML tags that convey the meaning and structure of content. These tags clearly define the role and purpose of the elements, making the content more understandable for browsers, search engines, and assistive technologies.

## **Importance for Accessibility and SEO:**

### **1. Accessibility:**

- Semantic tags make web content easier to navigate for users with disabilities who rely on screen readers or other assistive technologies.

### **2. SEO (Search Engine Optimization):**

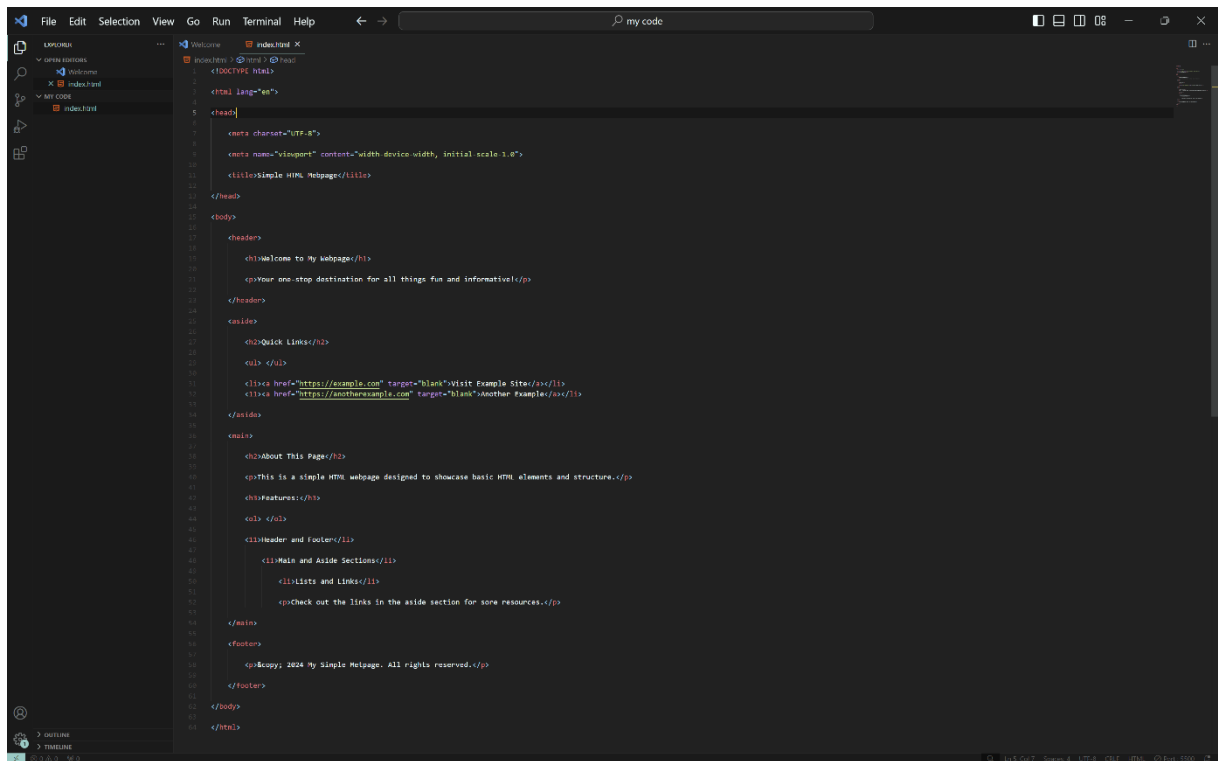
- Search engines use semantic tags to understand the structure and importance of content on a webpage.
- Proper use of semantic HTML improves search engine rankings and visibility.

## **Examples of Semantic Elements:**

1. `<header>`: Represents the introductory content or a section heading.
2. `<nav>`: Defines a navigation menu.
3. `<main>`: Highlights the primary content of a webpage.
4. `<section>`: Groups related content together.
5. `<article>`: Represents independent, self-contained content like blog posts or news articles.
6. `<footer>`: Contains footer information for the page or section.
8. `<aside>`: Represents content related to the main content, such as sidebars.

## **Lab Assignment:**

- Task: Create a simple HTML webpage that includes:
  - A header (), footer (), main section (), and aside section ().
  - A paragraph with some basic text.
  - A list (both ordered and unordered).
  - A link that opens in a new tab



```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>Simple HTML Mebpage</title>
7   </head>
8   <body>
9     <header>
10      <h1>Welcome to My Webpage</h1>
11      <p>Your one-stop destination for all things fun and informative.</p>
12    </header>
13    <aside>
14      <h2>Quick Links</h2>
15      <ul>
16        <li><a href="https://example.com" target="blank">Visit Example Site</a></li>
17        <li><a href="https://anotherexample.com" target="blank">Another Example</a></li>
18      </ul>
19    </aside>
20    <main>
21      <h2>About This Page</h2>
22      <p>This is a simple HTML webpage designed to showcase basic HTML elements and structure.</p>
23      <h3>Features:</h3>
24      <ul>
25        <li><ul>
26          <li><li>Header and Footer</li>
27          <li>Main and Aside Sections</li>
28          <li>Lists and Links</li>
29        </ul>
30        <li>Check out the links in the aside section for some resources.</li>
31      </ul>
32    </main>
33    <footer>
34      <p>©2024 My Simple Mebpage. All rights reserved.</p>
35    </footer>
36  </body>
37 </html>
```

## Output:



## Welcome to My Webpage

Your one-stop destination for all things fun and informative!

### Quick Links

- [Visit Example Site](#)
- [Another Example](#)

### About This Page

This is a simple HTML webpage designed to showcase basic HTML elements and structure.

#### Features:

- - Header and Footer
  - Main and Aside Sections
  - Lists and Links

Check out the links in the aside section for sore resources.

© 2024 My Simple Mebpage. All rights reserved.

## • HTML Forms

## Theory Assignment:

**Question 1: What are HTML forms used for? Describe the purpose of the input, textarea, select, and button elements?**

## HTML Forms:

HTML forms are used to collect and submit user input to a server. They enable interaction between the user and the web application, allowing data to be captured and processed for tasks like login, registration, feedback, or search.

### Purpose of Elements:

#### 1. <input>:

- o Used to collect user data like text, email, password, etc.
- o Example: `<input type="text" placeholder="enter your name ">`.

#### 2. <textarea>:

- o Used for multi-line text input, such as comments or feedback, it also provide attributes like rows and cols
- o Example: `<textarea placeholder="enter your message" rows="5" cols="3"></textarea>`

#### 3. <select>:

- o Creates a dropdown menu to choose from multiple options.
- o Example: `<select>`  
`<option value="1">option 1</option>`  
`<option value="2">option 2</option>`  
`</select>`

#### 4. <button>:

- o Triggers form submission or other actions when clicked.
- o Example: `<button type="submit">submit</button>`  
`<button type="reset">reset</button>`

**Question 2: Explain the difference between the GET and POST methods in form submission. When should each be used?**

	GET	POST
BACK button/Reload	Harmless	Data will be re-submitted (the browser should alert the user that the data are about to be re-submitted)
Bookmarked	Can be bookmarked	Cannot be bookmarked
Cached	Can be cached	Not cached
Encoding type	application/x-www-form-urlencoded	application/x-www-form-urlencoded or multipart/form-data. Use multipart encoding for binary data
History	Parameters remain in browser history	Parameters are not saved in browser history
Restrictions on data length	Yes, when sending data, the GET method adds the data to the URL; and the length of a URL is limited (maximum URL length is 2048 characters)	No restrictions
Restrictions on data type	Only ASCII characters allowed	No restrictions. Binary data is also allowed
Security	GET is less secure compared to POST because data sent is part of the URL  Never use GET when sending passwords or other sensitive information!	POST is a little safer than GET because the parameters are not stored in browser history or in web server logs
Visibility	Data is visible to everyone in the URL	Data is not displayed in the URL

## When be used

### GET Method:

- Used for retrieving data without modifying the server (e.g., search, filtering).
- Suitable for non-sensitive data and bookmarking URLs.

### POST Method:

- Used for submitting data that changes server data (e.g., login, form submission).
- Ideal for sensitive information and file uploads.

## Question 3: What is the purpose of the label element in a form, and how does it improve accessibility?

### Purpose of the element:

The label element is used to define a label for form controls (e.g. <input>, <textarea> , <select> , ). It links descriptive text to a specific form element, making the form easier to understand and interact with.

### How it improves accessibility:

1. Screen Reader Support: Screen readers can identify the purpose of the form control based on the associated label, improving usability for visually impaired users.

2. Clickable Area: When a is properly linked to a form element, clicking the label also activates the form control, enhancing usability.
3. Clear Context: Provides a clear and descriptive context for form fields, reducing confusion for all users.

### Lab Assignment:

**Task: Create a contact form with the following fields:**

- Full name (text input) • Email (email input)
- Phone number (tel input)
- Subject (dropdown menu)
- Message (textarea)
- Submit button

### Additional Requirements:

- Use appropriate form validation using required, minlength, maxlength, and pattern.
- Link form labels with their corresponding inputs using the for attribute

```
<!DOCTYPE html>
<html>
<body>

<h1>The label element</h1>

<p>Click on one of the text labels to toggle the related radio button:</p>

<form action="/action_page.php">
  <input type="radio" id="html" name="fav_language" value="HTML">
  <label for="html">HTML</label><br>
  <input type="radio" id="css" name="fav_language" value="CSS">
  <label for="css">CSS</label><br>
  <input type="radio" id="javascript" name="fav_language" value="JavaScript">
  <label for="javascript">JavaScript</label><br><br>
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

### Output:

## The label element

Click on one of the text labels to toggle the related radio button:

- ☐ HTML
- ☐ CSS
- ☐ JavaScript

Submit

### • HTML Tables Theory Assignment:

**Question 1: Explain the structure of an HTML table and the purpose of each of the following elements: `<table>`, `<tr>`, `<th>`, `<td>`, and `<thead>`?**

#### Structure of HTML Table:

An HTML table is a structured way to display tabular data in rows and columns. It uses a combination of specific elements to define the table's structure and content.

#### Purpose of Elements:

##### `<table>`:

- The `<table>` element is the container for all the table content.
- It defines the boundaries of the table and organizes its rows, columns, and content.

##### `<tr>` (Table Row):

- The `<tr>` element defines a single row within a table.
- It acts as a container for table cells (`<td>` or `<th>`).

##### `<th>` (Table Header):

- The `<th>` element defines a header cell in a table.
- It is typically used at the top of a column or the start of a row to label the data within that row or column.

##### `<td>` (Table Data):

- The <td> element represents a standard data cell within a table.
- It contains the actual data displayed in the rows and columns of the table.

#### **<thead> (Table Head):**

- The <thead> element groups one or more <tr> elements that define the header section of a table.
- It helps to semantically separate the table's header rows from its body and footer.

#### **Example of Table:**

```
<table border="1">
  <thead>
    <tr>
      <th>Product</th>
      <th>Price</th>
      <th>Quantity</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>samsung</td>
      <td>50,000</td>
      <td>1</td>
    </tr>
    <tr>
      <td>vivo</td>
      <td>75,000</td>
      <td>1</td>
    </tr>
  </tbody>
```



`</table>`

### Explanation of Example:

`<table>`: Defines the table.

`<thead>`: Groups the header row (`<tr>` with `<th>` elements).

`<tr>`: Creates rows.

`<th>`: Creates bold, centered header cells ("Product", "Price", "Quantity").

`<td>`: Defines the data cells ("Isamsung", "50,000", "1", "vivo", "75,000", "1").

### Question 2: What is the difference between colspan and rowspan in tables? Provide examples?

#### Difference Between colspan and rowspan:

##### 1. colspan:

- Merges a cell across multiple columns.
- Expands a single cell horizontally to occupy the space of two or more columns.
- When a single value needs to cover multiple columns, such as in table headings or when grouping data.

##### 2. rowspan:

- Merges a cell across multiple rows.
- Expands a single cell vertically to occupy the space of two or more rows.
- When a single value applies to multiple rows, such as a common label or grouping.

```
<table>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
  <tr>
    <td>February</td>
    <td>$100</td>
  </tr>
  <tr>
    <td colspan="2">Sum: $180</td>
  </tr>
</table>
```

Output:

Month	Savings
January	\$100
February	\$80
Sum: \$180	

**Question 3: Why should tables be used sparingly for layout purposes? What is a better alternative?**

**Why Tables Should Be Used Sparingly for Layout Purposes:**

- 1. Lack of Flexibility:** Tables are rigid and not responsive, making it hard to adapt to different screen sizes and devices.
- 2. Accessibility Issues:** Using tables for layout confuses screen readers, which are designed to interpret tables as data structures. This can hinder accessibility for users with disabilities.
- 3. Code Complexity:** Tables for layout require excessive HTML, making the code harder to read, maintain, and debug.
- 4. Poor SEO:** Search engines may misinterpret table layouts, negatively impacting rankings.

**Better Alternative: CSS for Layout**

**CSS (Cascading Style Sheets)** is a more efficient and modern way to create layouts. It provides flexibility, responsiveness, and accessibility.

**Recommended CSS Techniques:**

**1. Flexbox:**

- o Ideal for creating one-dimensional layouts (e.g., rows or columns).
- o Example:

```
<div style="display: flex;">  
  <div>Column 1</div>  
  <div>Column 2</div> </div>
```

## 2. CSS Grid:

o Best for two-dimensional layouts, such as grids or complex designs. o Example:

```
<div style="display: grid; grid-template-columns:1fr 1fr;">  
  <div>Item 1</div>  
  <div>Item 2</div>  
</div>
```

### Conclusion:

Instead of tables, use CSS Flexbox or CSS Grid for creating layouts. They are more responsive, accessible, and maintainable while adhering to modern web development standards.

### Lab Assignment:

• **Task: Create a product catalog table that includes the following columns:**

• **Product Name**

• **Product Image (use placeholder image URLs)**

• **Price**

• **Description**

• **Availability (in stock, out of stock)** **Additional Requirements:**

• **Use thead for the table header.**

• **Add a border and some basic styling using inline CSS.**

• **Use colspan or rowspan to merge cells where applicable.**

```
File Edit Selection View Go Run Terminal Help my code
<!--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>
    <h1>Product Catlog</h1>
    <table border="1" cellpadding="8" cellspacing="10" style="border-collapse: collapse; width:100%; text-align: left;">
      <thead style="background-color: #eafaf4;">
        <tr>
          <th>Product Name</th>
          <th>Product Image</th>
          <th>Price</th>
          <th>Description</th>
          <th>Availability</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <td rowspan="2">Product a</td>
          <td rowspan="2" style="text-align: center;">
            
          </td>
          <td>30$</td>
          <td>It Is Best For Man</td>
          <td>In Stock</td>
        </tr>
        <tr>
          <td colspan="4" style="text-align: center; background-color: #e9f0f9;">
            Buy 1 Get 1 Free
          </td>
        </tr>
        <tr>
          <td>Product b</td>
          <td style="text-align: center;">
            
          </td>
          <td>1500$</td>
          <td>It Is Good For Photography</td>
          <td>In Stock</td>
        </tr>
        <tr>
          <td>product c</td>
          <td style="text-align: center;">
            
          </td>
          <td>50$</td>
          <td>It is classic</td>
          <td>Out Of Stock Shortly</td>
        </tr>
      </tbody>
    </table>
  </body>
</html>
```

Output:



Product Catlog

Product Name	Product Image	Price	Description	Availability
Product a		30\$	It Is Best For Man	In Stock
		Buy 1 Get 1 Free		
Product b		1500\$	It Is Good For Photography	In Stock
product c		50\$	It is classic	Out Of Stock Shortly

