

Module 15) HTML in Full Stack

1. HTML Basics

Theory Assignment

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

HTML stands for HyperText Markup Language. It is the standard language used to create web pages and web applications. HTML is not a programming language. It is a markup language that tells the web browser how to display content. Web pages are made up of different elements like headings, paragraphs, images, links, lists, tables, and forms, and HTML is used to define all these elements. The main purpose of HTML in web development is to structure the content of a website. It allows developers to organize text, images, and other media in a way that browsers can understand. HTML also makes it possible to create links that connect different pages on the web. It helps in embedding images, videos, and audio so that web pages can be more interactive and engaging. Forms in HTML allow websites to collect information from users, such as names, email addresses, and feedback. HTML works together with CSS, which is used for styling, and JavaScript, which is used to make web pages interactive. Without HTML, there would be no structure to websites, and browsers would not know how to display content properly. It is the backbone of all websites.

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

An HTML document has a basic structure that all web pages follow. It starts with the `<!DOCTYPE html>` declaration, which tells the browser that the page is written in HTML5. The HTML content is enclosed within `<html>` tags, which act as the root of the document. Inside the `<html>` tags, there are two main sections: `<head>` and `<body>`.

The `<head>` section contains information about the web page that is not displayed directly on the screen. This includes the title of the page, which is set using the `<title>` tag, links to stylesheets, metadata like keywords and descriptions, and scripts if needed. The `<title>` tag is mandatory because it provides the name of the page shown in the browser tab.

The `<body>` section contains all the content that is visible to users, such as text, images, links, lists, tables, and forms. Anything that should appear on the web page goes inside the `<body>` tags.

The mandatory tags in a basic HTML document are `<!DOCTYPE html>`, `<html>`, `<head>`, `<title>`, and `<body>`. `<!DOCTYPE html>` declares the HTML version, `<html>` defines the root of the document, `<head>` contains metadata, `<title>` sets the page title, and `<body>` holds the visible content. Together, these tags form the basic structure of every HTML page.

Question 3: What is the difference between block-level elements and inline elements in HTML? Provide examples of each.

In HTML, elements are classified as block-level or inline based on how they are displayed on a web page. Block-level elements take up the full width of their parent container and always start on a new line. They create a “block” of content. Examples of block-level elements include `<div>`, `<p>`, `<h1>` to `<h6>`, ``, ``, and ``. For instance, if you use two `<p>` tags one after another, each paragraph will appear on a separate line with space between them.

Inline elements, on the other hand, do not start on a new line. They only take up as much width as necessary for their content and flow along with other inline elements within the same line. Examples of inline elements include ``, `<a>`, ``, ``, and ``. For example, if you have text inside a `<p>` tag and use `` for some words, those words will appear bold but stay on the same line as the surrounding text.

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

Provide examples of semantic elements.

Semantic HTML uses elements that clearly describe their meaning and purpose in a web page, both for the browser and for humans reading the code. Unlike non-semantic tags like `<div>` and ``, semantic elements convey the type of content they contain. Examples include `<header>` for the top section of a page, `<nav>` for navigation links, `<main>` for the main content, `<article>` for independent content, `<section>` for different sections, `<aside>` for side content, and `<footer>` for the bottom section.

The role of semantic HTML is important for accessibility because assistive technologies, like screen readers, can understand and navigate the content more easily. For example, a screen reader can jump directly to the `<main>` content or navigate through `<nav>` links, helping users with disabilities interact with the page efficiently.

Semantic HTML is also important for SEO (Search Engine Optimization) because search engines can better understand the structure and meaning of the content. This helps pages rank higher in search results and ensures that relevant content is indexed correctly.

Lab Assignment

Task: Create a simple HTML webpage that includes:

- o A header (`<header>`), footer (`<footer>`), main section (`<main>`), and aside section (`<aside>`).**
- o A paragraph with some basic text.**
- o A list (both ordered and unordered).**
- o A link that opens in a new tab.**

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>HTML Assignment</title>
</head>
<body>
  <header>
    <h1>HEADER SECTION</h1>
    <nav>
      <ul>
        <li><a href="#">Home</a></li>
        <li><a href="#">About</a></li>
        <li><a href="#">Contact</a></li>
      </ul>
    </nav>
  </header>
  <main>
    <h1>Main Section</h1>
    <p>this is simple paragraph and used to know about the html tags</p>
    <aside>
      <h3>Related Links</h3>
      <ul>
        <li><a href="#">Link 1</a></li>
        <li><a href="#">Link 2</a></li>
      </ul>
    </aside>
  </main>
  <footer>
    <p>&copy; 2025 Example Company</p>
  </footer>
</body>
</html>
```

2. 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 are used to collect input from users on a web page, such as names, emails, feedback, or search queries. The `<input>` element allows users to enter data like text, passwords, or numbers. `<textarea>` provides a larger area for multi-line text input. `<select>` creates a dropdown menu so users can choose from multiple options. The `<button>` element is used to submit the form or trigger actions when clicked. Together, these elements make forms interactive and allow websites to receive and process user information.

Question 2: Explain the difference between the GET and POST methods in form submission.

When should each be used?

In HTML forms, the GET and POST methods determine how form data is sent to the server.

The GET method appends form data to the URL as query parameters. It is visible in the browser's address bar and has a limited length. GET is suitable for retrieving data or performing actions that do not change the server's state, like search forms.

The POST method sends form data in the request body, which is not visible in the URL and can handle larger amounts of data. POST is used for actions that modify server data, like submitting registration forms, login credentials, or uploading files.

In short, GET is for safe, visible requests, while POST is for secure or large data submissions.

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

The `<label>` element in a form is used to define a text description for an input field, such as a text box, checkbox, or radio button. Its main purpose is to tell users what information is expected in the input.

Using `<label>` improves accessibility because screen readers can read the label aloud, helping visually impaired users understand the purpose of the input field. Labels also increase the clickable area for form controls, making it easier for all users to select checkboxes or radio buttons.

Lab Assignment

Task: Create a contact form with the following fields:

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

Additional Requirements:

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

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <title>Contact Form</title>
```

```
</head>
```

```
<body>
```

```
  <h1>Contact Us</h1>
```

```
  <form action="#" method="post">
```

```
    <label for="full-name">Full Name:</label>
```

```
    <input type="text" id="full-name" name="full-name" required minlength="2" maxlength="50">
```

```
    <label for="email">Email:</label>
```

```
    <input type="email" id="email" name="email" required>
```

```
    <label for="phone">Phone Number:</label>
```

```
<input type="tel" id="phone" name="phone" pattern="[0-9]{10}" required>
```

```
<label for="subject">Subject:</label>
```

```
<select id="subject" name="subject" required>
```

```
  <option value="" disabled selected>Select a subject</option>
```

```
  <option value="general-inquiry">General Inquiry</option>
```

```
  <option value="technical-support">Technical Support</option>
```

```
  <option value="billing">Billing</option>
```

```
</select>
```

```
<label for="message">Message:</label>
```

```
  <textarea id="message" name="message" required minlength="10"
maxlength="500"></textarea>
```

```
<button type="submit">Submit</button>
```

```
</form>
```

```
</body>
```

```
</html>
```

3. 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>.

An HTML table is used to display data in rows and columns. The <table> element defines the start and end of the table. Inside <table>, <tr> (table row) is used to create each row. <th> (table header) defines header cells, usually displayed in bold and centered, which describe the content of the columns. <td> (table data) defines standard cells that hold the actual data in the table. <thead> groups the header rows of a table, helping browsers and assistive technologies identify the headings separately from the main content. Together, these elements organize data clearly and accessibly.

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

In HTML tables, colspan and rowspan are used to make a cell span across multiple columns or rows.

Colspan allows a cell to extend horizontally across several columns. For example, <td colspan="3"> will make that cell occupy three columns in the same row.

Rowspan allows a cell to extend vertically across multiple rows. For example, <td rowspan="2"> will make the cell cover two rows in the same column.

In short, colspan stretches a cell across columns, while rowspan stretches a cell across rows, helping create more flexible table layouts.

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

Tables should be used sparingly for layout because they are meant to display tabular data, not design the structure of a page. Using tables for layout makes the HTML code complex, harder to maintain, and less accessible for screen readers or search engines. It also reduces flexibility when adapting the page for different screen sizes, like on mobile devices.

A better alternative is to use CSS with elements like <div>, <section>, and modern layout techniques such as Flexbox or CSS Grid. These methods provide more control, cleaner code, and responsive designs while keeping content and presentation separate.

Lab Assignment

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

- o Product Name
- o Product Image (use placeholder image URLs)
- o Price
- o Description
- o Availability (in stock, out of stock)

Additional Requirements:

- o Use thead for the table header.
- o Add a border and some basic styling using inline CSS.
- o Use colspan or rowspan to merge cells where applicable.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Product Catalog</title>
</head>
<body>

  <table style="border: 1px solid black; border-collapse: collapse;">
    <thead style="background-color: lightgray;">
      <tr>
        <th style="border: 1px solid black; padding: 8px;">Product Name</th>
        <th style="border: 1px solid black; padding: 8px;">Product Image</th>
        <th style="border: 1px solid black; padding: 8px;">Price</th>
        <th style="border: 1px solid black; padding: 8px;">Description</th>
        <th style="border: 1px solid black; padding: 8px;">Availability</th>
      </tr>
```



```

</thead>

<tbody>

  <tr>

    <td style="border: 1px solid black; padding: 8px;">Laptop Pro</td>

    <td style="border: 1px solid black; padding: 8px;"></td>

    <td style="border: 1px solid black; padding: 8px;">$1200.00</td>

    <td style="border: 1px solid black; padding: 8px;">A high-performance laptop for
professionals.</td>

    <td style="border: 1px solid black; padding: 8px;">In Stock</td>
  </tr>

  <tr>

    <td style="border: 1px solid black; padding: 8px;">Wireless Mouse</td>

    <td style="border: 1px solid black; padding: 8px;"></td>

    <td style="border: 1px solid black; padding: 8px;">$25.00</td>

    <td style="border: 1px solid black; padding: 8px;">Ergonomic mouse with long battery
life.</td>

    <td style="border: 1px solid black; padding: 8px;">In Stock</td>
  </tr>

  <tr>

    <td style="border: 1px solid black; padding: 8px;">Headphones</td>

    <td style="border: 1px solid black; padding: 8px;"></td>

    <td style="border: 1px solid black; padding: 8px;">$90.00</td>

    <td style="border: 1px solid black; padding: 8px;">Noise-cancelling headphones for
immersive audio.</td>

    <td style="border: 1px solid black; padding: 8px;" rowspan="2">Out of Stock</td>
  </tr>

  <tr>

```

`<td style="border: 1px solid black; padding: 8px;" colspan="4">Special Offer: All audio products are temporarily unavailable.</td>`

`</tr>`

`</tbody>`

`</table>`

`</body>`

`</html>`