HTML Basics

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

**HTML (HyperText Markup Language)** is the standard language used to create and structure webpages. It defines the content of a webpage using elements (tags) that specify headings, paragraphs, links, images, tables, and more.

**Purpose in Web Development:**

* Provides **structure** to a webpage (like a blueprint).
* Works with **CSS (for styling)** and **JavaScript (for interactivity)** to create functional and visually appealing websites.
* Allows browsers to **interpret and display** content correctly.

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

A basic HTML document consists of the following essential tags:

<!DOCTYPE html>

<head>

<meta charset="UTF-8">

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

<title>Page Title</title>

</head>

<body>

<h1> Main heading </h1>

<p> Page content.</p>

</body>

</html>

**Mandatory Tags & Their Purposes:**

1. <!DOCTYPE html> – Declares the document as an HTML5 file.
2. <html> – The root element containing all webpage content.
3. <head> – Stores metadata (like title, character encoding, and styles).
4. <title> – Defines the webpage title shown in the browser tab.
5. <body> – Contains all the visible content (text, images, links, etc.).

**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 available.
* Typically used for major content sections.

**Examples:**

* <div> – Generic container
* <p> – Paragraph
* <h1> to <h6> – Headings
* <ul>, <ol>, <li> – Lists
* <section>, <article>, <footer>, <header> – Semantic layout elements

**Inline Elements:**

* Stay within the same line as surrounding content.
* Typically used for styling or linking small parts of text.

**Examples:**

* <span> – Generic inline container
* <a> – Hyperlink
* <strong> – Bold text
* <em> – Italic text
* <img> – Image
* <code> – Inline code snippet

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

**Semantic HTML** refers to using meaningful tags that **describe the content's purpose**, improving accessibility and SEO.

**Importance:**

1. **Accessibility:**
   * Helps screen readers and assistive technologies understand content better.
   * Makes navigation easier for users with disabilities.
2. **SEO (Search Engine Optimization):**
   * Search engines prioritize well-structured content.
   * Improves ranking and discoverability.
3. **Readability & Maintainability:**
   * Easier for developers to understand and modify the code.

**Examples of Semantic Elements:**

* <header> – Represents the introductory section or navigation.
* <nav> – Defines navigation links.
* <main> – Represents the main content of the document.
* <article> – Self-contained piece of content (e.g., a blog post).
* <section> – Groups related content together.
* <aside> – Sidebar content related to the main content.
* <footer> – Contains footer information (e.g., copyright, contact details).

HTML Forms

**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 user input and send data to a server for processing. Forms allow users to enter information such as text, passwords, choices, or files.

**Key Form Elements & Their Purposes:**

1. **<input>** – Allows users to enter data in various formats, such as text, passwords, emails, and numbers.
2. <input type="text" name="username" placeholder="Enter your name">
3. **<textarea>** – Provides a multi-line text input field, useful for comments or messages.
4. <textarea name="message" rows="4" cols="30">Type your message...</textarea>
5. **<select>** – Creates a dropdown list for users to select an option.
6. <select name="country">
7. <option value="us">United States</option>
8. <option value="uk">United Kingdom</option>
9. </select>
10. **<button>** – A clickable button to submit or reset a form.
11. <button type="submit">Submit</button>

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

1. **GET Method:**
   * Appends form data to the **URL** as query parameters.
   * Used for **retrieving** data (e.g., search queries).
   * Less secure since data is visible in the URL.
   * Example:
   * <form action="search.php" method="GET">
   * <input type="text" name="query">
   * <button type="submit">Search</button>
   * </form>
   * Best for **search forms** or **bookmarkable requests**.
2. **POST Method:**
   * Sends form data in the **request body** (not visible in the URL).
   * Used for **submitting sensitive or large data** (e.g., login, registration).
   * More secure and supports **file uploads**.
   * Example:
   * <form action="submit.php" method="POST">
   * <input type="text" name="name">
   * <button type="submit">Submit</button>
   * </form>
   * Best for **login forms, payment forms, or updating database records**.

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

The <label> element is used to associate text with a form control, improving usability and accessibility.

**Benefits:**

* Helps **screen readers** identify form fields.
* Increases **clickability** (users can click the label to focus the input).
* Enhances **form navigation** for keyboard users.

**Example:**

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

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

HTML Tables

**Question 1: Explain the structure of an HTML table and the purpose of each element (<table>, <tr>, <th>, <td>, and <thead>).**

An HTML table organizes data in rows and columns.

**Key Table Elements & Their Purposes:**

1. **<table>** – Defines the table structure.
2. **<tr> (Table Row)** – Represents a row in the table.
3. **<th> (Table Header Cell)** – Defines a header cell (bold, centered by default).
4. **<td> (Table Data Cell)** – Defines a standard table cell.
5. **<thead>** – Groups header rows for better structuring.

**Example:**

<table border="1">

<thead>

<tr>

<th>Name</th>

<th>Age</th>

</tr>

</thead>

<tr>

<td>Alice</td>

<td>25</td>

</tr>

</table>

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

* **colspan**: Merges multiple columns into a single cell.
* **rowspan**: Merges multiple rows into a single cell.

**Example:**

<table border="1">

<tr>

<th colspan="2">Full Name</th>

<th>Age</th>

</tr>

<tr>

<td>John</td>

<td>Doe</td>

<td>30</td>

</tr>

<tr>

<td rowspan="2">Alice</td>

<td>Smith</td>

<td>25</td>

</tr>

<tr>

<td>Brown</td>

<td>28</td>

</tr>

</table>

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

**Problems with Using Tables for Layout:**

* **Not responsive** – Tables do not adapt well to different screen sizes.
* **Difficult to maintain** – Updating the layout requires complex modifications.
* **Poor accessibility** – Screen readers may struggle to interpret a table-based layout.
* **SEO issues** – Search engines prefer semantic, well-structured content.

**Better Alternative:**  
Use **CSS with divs and flexbox/grid** to create layouts.

**Example (Flexbox Layout):**

<div style="display: flex;">

<div style="flex: 1; background: lightblue; padding: 20px;">Column 1</div>

<div style="flex: 1; background: lightgreen; padding: 20px;">Column 2</div>

</div>