1. HTML Basics

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

ANS :-

- HTML Stands for Hypertext Markup language.
- HTML is the standard markup language for creating web pages.
- HTML describes the structure of web pages.

Purpose of HTML in web development :-

- It provides the basic structure of webpage.
- It defines content and layout that web browsers can display.
- It allows integration of text, images , videos and other media.
- Use hyperlinks to connect different web pages or resources.
- It works with CSS(for styling) and JavaScript(for interactivity) to build complete web Applications.

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

ANS :-

The basic structure of an HTML document includes the <!DOCTYPE>, <html>, <head>, <title>, and <body> tags.

```
<!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>

  <p>This is HTML Structure</p>

  <h1>This is Heading tag</h1>
```

```
</body>

</html>
```

Tags	Purpose
<!DOCTYPE html>	Declaration defines that this document is HTML5 Document.
<html>	The <html> element is the root element of an HTML page
<head>	The <head> element contains meta information about the HTML page (e.g., title, character set, links to style sheets, etc.).
<meta charset="UTF-8">	Sets the character encoding (important for text display).
<title>	The <title> element specifies a title for the HTML page or page of title (which is shown in the browser's title bar or in the page's tab)
<body>	The <body> element defines the document's body, and is a container for all the visible contents, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.

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

ANS :-

Block-level Elements :-

- A block-level elements always starts on a new line.
- The browser automatically add some space before and after the element.
- Always takes up the full width available.
- Two commonly used block elements are: <p> and .

Example :-

```
<P>Hello world</p>
```

```
<div>Hello world</div>
```

Inline Elements :-

- An inline elements does not start on a new line.
- An inline element only takes up as much width as necessary.
- This is a element inside a paragraph.

Example :-

```
<span>Hello World</span>
```

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

ANS :- Semantic HTML means using HTML elements that have a meaningful name and purpose, describing their role in the content rather than just their appearance.

- **Role of Semantic HTML :**

Structure & Readability : Makes your code easier for developers to understand.

Accessibility : Screen readers can navigate content correctly.

SEO (Search Engine Optimization) : Search engines better understand your page content and importance.

Maintainability : Easier to style and modify without breaking meaning.

- **Important :**

Accessibility : Screen readers rely on semantic tags to tell visually impaired users what section they're in.

SEO : Search engines like Google use semantic elements to understand: Which part is a heading is the main article content. Which is navigation or footer

Example :

```
<h1>My Blog</h1>

<footer>

<p>&copy; 2025 My Blog</p>

</footer>
```

2. HTML Forms

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

ANS :-

- An html form is a section of a document which contains controls such as text fields , passwords , checkboxes , radio , submit , menus etc.
- An html form is used to collect user input.
- The user input is most often sent to a server for processing.

Key HTML Form Elements & Their Purposes

<input> :- The <input> element can be displayed in several ways , depending on the type attribute.

Example :

```
<input type="text" name="username" placeholder="Enter your name">

<input type="email" name="email" placeholder="Enter your email">
```

<textarea> :-The <textarea> element defines a multiline input field.

Example :

```
<textarea name="message" rows="5" cols="30">Your message here</textarea>
```

<select> :- The <select> element defines the drop-down list. The <option> element defines an option that can be selected.

Example:

```
<select name="country">

        <option value="india">India</option>

         <option value="usa">USA</option>

</select>
```

<button> :- The <button> element defines a clickable button.

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?

ANS :-HTTP methods defines the specific action that can be performed on the resources like java or servlet file.

GET	POST
GET method sends data through the resources URL and thus is not secured.	POST method sends data through the http message body and thus it is more secured.
GET is slightly faster because the values are sent in the header.	POST is slightly slow because the values are sent in the format that the content type specifies.

Less data store and access.	More and large data stored and acess.
Get request can be cached.	POST request cannot be cached.
Get request can be bookmarked.	Post method cannot be bookmarked.

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

ANS :-

- The <label> elements defines a label for several form elements.
- The for attribute of the <label> tag shouls be equal to the id attribute of the <input> element to bind them together.

Purpose of the <label> Element

- The <label> element in HTML is used to provide a text description for form controls such as <input>, <textarea>, <select>, etc.

How does it improve accessibility :-

- **Helps Screen Readers**
  Screen readers read the label aloud, helping visually impaired users understand what to enter.
- **Clickable Area**
  Clicking on a label selects the linked input (e.g., checkbox), making forms easier to use.
- **Keyboard Friendly**
  Labels help keyboard-only users know what each field is for.
- **Better Form Clarity**
  Makes forms clearer and more user-friendly for everyone.

3. HTML Tables

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

ANS :-

- HTML table allow web developers to arrange data into rows and columns.
- A table in html consists of table cells inside rows and columns.

Structure of HTML table :-

```
<table>
 <thead>
  <tr>
   <th>Heading 1</th>
   <th>Heading 2</th>
  </tr>
 </thead>
 <tr>
  <td>Data 1</td>
  <td>Data 2</td>
 </tr>
 <tr>
  <td>Data 3</td>
  <td>Data 4</td>
 </tr>
</table>
```

Purpose of all elements :-

<table> :- The main container that defines the start and end of the table. All rows and data go inside this tag.

<tr> :- stands table row. Defines a row in a table.

<th> :- stands table header. Defines a table header cell in a table.

<td> :- stands table data. Defines a table data cell in a table.

<thead> :- Groups one or more rows of header content () at the top of the table.

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

ANS :- The colspan and rowspan attributes in HTML tables are used to merge cells across multiple columns or rows, respectively. They are applied to or elements.

Colspan :- to make a cell span over multiple columns .merges to multiple columns into one cell.

Rowspan :- merges to multiple rows into one cell.

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

ANS :-

- Tables should not be used much for webpage layouts because they were originally made to show data like rows and columns, not to design how a webpage looks. If we use tables for layout, a few problems come up:
- **Hard for disabled users –** Screen readers (used by blind people) may read the page in a confusing way because they expect a table of data, not a design layout.
- **Not mobile-friendly –** Tables don't easily shrink or rearrange for small screens, so the page can look messy on phones.
- **Difficult to edit –** If you want to change the design later, you'll need to mess with the table structure, which can be frustrating and time-consuming.

Better alternative :-

- A much better way is to use CSS (Cascading Style Sheets).
- With CSS, you can control how things look without messing up meaning. Tools like Flexbox and CSS Grid make it easy to build layouts that are responsive work well on phones, tablets, and computers and easy to update.