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 and design **web pages**.

Purpose of HTML in Web Development:

- It gives structure to a web page.
- It is **not a programming language**, but a **markup language**. This means HTML is used to **mark** different parts of a webpage like: **text**, **headings, Paragraphs, images**, **links**, **buttons, Tables, Forms, Videos and More**.
- HTML gives the **basic structure** to a webpage. Just like a building needs a framework, a website needs HTML.
- Work together with other languages
  1. HTML gives the **structure**
  2. CSS gives the **design**
  3. JavaScript adds **functionality**

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

```
<!DOCTYPE html>
<html>
 <head>
  <title>Page Title</title>
 </head>
 <body>
```

```
    <h1>This is a Heading</h1>
    <p>This is a paragraph. </p>
  </body>
</html>
```

Explanation of Mandatory Tags:

Tag	Purpose (Simple Explanation)
<!DOCTYPE html>	Tells the browser that this is an HTML5 document.
<html>	The root of the webpage. It wraps all the content.
<head>	Contains **information about the page**, like title, styles, etc. (not shown on the page)
<title>	Sets the title of the page (shows on the browser tab).
<body>	Contains all the content you want to show on the page (text, images, links, etc.)

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

1. Block-level Elements:

- They **take up the full width** of the page (from left to right).
- They **start on a new line**.
- Used to create **larger sections** of a webpage like paragraphs, headings, divs etc.
- They can have **other elements inside** them.

✅ **Examples of block-level elements:**

- This is a block
- This is a paragraph
- <h1>This is a heading</h1>

2. Inline Elements:

- They **only take as much space** as they need.
- They **don't start on a new line.**
- Used to style or format **small parts of text** within a line, like links, bold text, or spans.
- They stay in the **same line as the text**.

✅ **Examples of inline elements:**

- This is a span
- This is a link
- Bold text

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

- Semantic HTML means using HTML tags that **describe the meaning** of the content — not just how it looks.

  For example:
  - means a paragraph
  - <header> means the top section
  - <nav> means navigation (menu)
  - <main> Main content of the page
  - <section> A part or block of content
  - <article> Standalone content (like blog)
  - <footer> Bottom part (contact, copyright)
  - <aside> Sidebar or related info

Accessibility :

Helps screen readers read the page for visually impaired users. Semantic tags like <nav>, <main>, and <header> make it easier for them.

Seo :

Search engines like Google understand the structure better with semantic tags, which helps websites rank higher.

- **Simple Example:**

```
<article>
  <header>
    <h1>My Blog Post</h1>
  </header>
  <p>This is the content of my blog.</p>
  <footer>Written by John</footer>
</article>
```

Question 5: 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** — like names, emails, passwords, choices, feedback, etc.
  For example: login forms, contact forms, surveys, Giving feedback etc.
- An **HTML form** is a part of a web page that allows **users to enter data** and **send it to the server**.
- There are **Two Submission Methods** of HTML Form.
  1. GET Method
  2. POST Method

Common Form Elements and Their Purpose

1. <input>

Used to take single-line input from the user like:

- Name
- Email

- Password
- Checkbox
- Radio button

  And more (using type attribute)

```
<input type="text" placeholder="enter your name">
```

2. <textarea>

Used to take **multi-line input** like feedback or comments.

```
<textarea placeholder="Write your comments here"></textarea>
```

3. <select>

Used to create a **dropdown list** where users can select one (or more) options.

```
<select>
  <option value="gujarat">Gujarat</option>
  <option value="maharashtra">Maharashtra</option>
</select>
```

4. <button>

Used to create a button, usually for **submitting** the form or doing some action.

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

Question 6: Explain the difference between the GET and POST methods in form submission. When should each be used?

Difference Between Methods :

1. GET Method

- Sends data in the **URL**
- Not secure (visible to everyone)
- Used when data is not sensitive (like search)
- <form action="search.php" method="get">
- Parameter are saved in browser history
- Support only string data types
- Has a length limitation of 255 Characters

2. POST Method

- Sends data **securely in the background**
- Not shown in URL
- Used for passwords, login, sign-up, contact forms etc.
- <form action="login.php" method="post">
- Parameter are saved in browser history
- Support Different data type Such as, String, numeric, binary etc.
- Does not have a length limitation

Question 7: What is the purpose of the label element in a form, and how does it improve accessibility?

- The <label> tag is used to **name or describe a form field** like an input box, checkbox, radio button, etc.
- It tells the user **what to enter** in that field.
- Improves form usability and user experience
- Most importantly: improves **accessibility**
  1. The <label> tells screen readers what each input field is for.
  2. This helps **blind or visually impaired users** know what to enter.

Without <label>:

- A screen reader might just say "**edit box**" — unclear.
- The user doesn't know **what information to enter.**

With <label>:

- A screen reader will say "**Name: edit box**"
- Now the user knows: "Oh, I have to enter my name here."

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

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

Question 8: Explain the structure of an HTML table and the purpose of each of the following elements: <table>,<tr>,<td>,<th> and <thead>.

- HTML Table is used to display data in tabular form(Rows & Columns).
- There can be used many columns in a row.
- HTML Tables are used to manage the layout of the page.
  Example : Header section, Navigation section, Body Content ,Footer section etc.
- An HTML Table is created using the <table>tag. inside this tag we use <tr>,<th>,<td>.

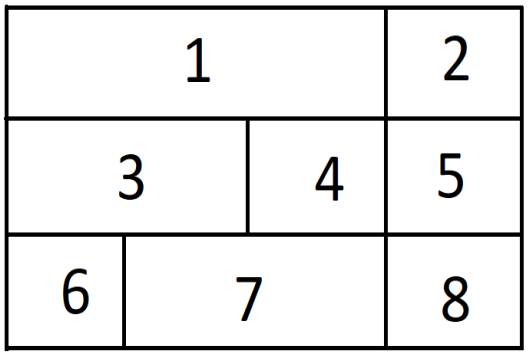
Purpose Of HTML Table elements :

Tag	Role in the Example
<table>	Starts and ends the whole table
<thead>	Groups the heading row (<tr> with <th>)
<tr>	Creates a new row (1 for headings, others for data)
<th>	Table heading cells: “Name”, “Age”
<td>	Table data cells: “John”, “25”, etc.

Question 9: What is the difference between colspan and rowspan in tables? Provide examples.

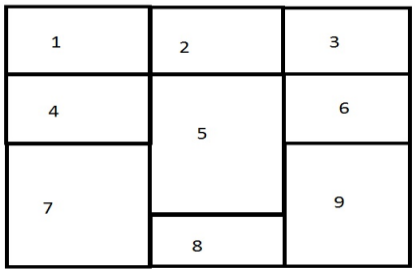
1.Colspan :

- Use colspan to **merge cells left to right**
- Works **horizontally**(left to right)
- Merging multiple **columns** into one
- Usually in <th> or <td>



2.Rowspan :

- Use rowspan to **merge cells top to bottom** **1,9,7**
-
- Works **vertically** (top to bottom)
- Merging multiple **rows** into one
- Usually in <th> or <td>



Question 10: Why should tables be used sparingly for layout purposes? What is a better alternative?

- **Why tables should be used sparingly for layout:**
  - They make the web page slower
  - The code becomes big and hard to understand
  - It's not good for mobile or responsive design (doesn't adjust well to different screen sizes)
  - It's harder for screen readers (less accessible)
- **Better alternative:**
  - Use **CSS layout** like **Flexbox** or **CSS Grid**

- This makes the webpage clean, fast, and responsive

In short: Use tables only for showing **data**, and use **CSS (Flexbox/Grid)** for page layout.