1. What is a Website?

A website is a collection of web pages that users can access through the internet using a web browser. Websites are categorized into:

- **Static Websites:** Simple pages with fixed content (e.g., portfolio websites).
- **Dynamic Websites:** Websites that update content dynamically (e.g., social media platforms, e-commerce sites).

A website consists of three core parts:

A. Frontend (Client-Side)

- **HTML (HyperText Markup Language):** The structure of a webpage.
- **CSS (Cascading Style Sheets):** Styling and layout.
- **JavaScript:** Adds interactivity and dynamic behavior.

B. Backend (Server-Side)

- The logic and database management behind the website.
- Backend languages: **Node.js, Python, PHP, etc.**

C. Database

- Stores user data, transactions, and other content.
- Examples: **MongoDB, MySQL, PostgreSQL.**

---

2. How a Website Loads: Step-by-Step Process

Step 1: User Requests a Web Page

- The user enters a URL (e.g., www.example.com) in their browser.
- The browser contacts the **DNS (Domain Name System)** to get the website's IP address.
- An **HTTP/HTTPS request** is sent to the web server.

Step 2: Server Processes the Request

- If it's a **static website**, the server sends back an HTML file.
- If it's a **dynamic website**, the server runs backend code, fetches data from the database, and then generates an HTML response.

Step 3: Browser Receives and Renders the Page

- The browser downloads the **HTML, CSS, and JavaScript** files.
- It processes them in the following order:
    1. **Parses HTML** and builds the **DOM (Document Object Model)**.

2. **Processes CSS** and creates the **CSSOM (CSS Object Model).**
3. **Combines DOM and CSSOM** to form the **Render Tree.**
4. **Executes JavaScript** to modify the page dynamically.
5. **Paints the final webpage** on the screen.

Section 1: HTML Basics

Chapter 1: Introduction to HTML

1. What is HTML?

HTML (**HyperText Markup Language**) is the standard language used to create web pages. It defines the structure and content of a webpage using **tags and elements**.

Key Points:

- HTML is not a programming language; it is a **markup language** used for structuring web content.
- It consists of a series of **elements** that tell the browser how to display content.
- HTML works alongside **CSS (for styling)** and **JavaScript (for interactivity).**

Example of an HTML element:

```
<p>This is a paragraph.</p>
```

---

2. Role of HTML in Web Development

HTML forms the **backbone of web development** by structuring content on the web. It provides:

- **Text Formatting:** Headings, paragraphs, lists, and links.
- **Media Embedding:** Images, videos, and audio.
- **Forms and Input Fields:** Collecting user data.
- **Navigation & Layout:** Links, sections, and divs to organize content.

How HTML Works with CSS & JavaScript:

- **HTML** provides the structure.
- **CSS** styles the content (colors, fonts, layout).
- **JavaScript** adds interactivity (buttons, animations, user interactions).

---

3. Structure of an HTML Document

Every HTML document follows a **basic structure** with essential tags:

Basic HTML Template:

```
<!DOCTYPE html>
<html>
<head>
    <title>My First Webpage</title>
</head>
<body>
    <h1>Welcome to My Website</h1>
    <p>This is my first webpage using HTML.</p>
</body>
</html>
```

Explanation of Tags:

1. <!DOCTYPE html> → Defines the document type and version of HTML.
2. <html> → The root element that wraps all content.
3. <head> → Contains metadata (title, styles, scripts, etc.).
4. <title> → Sets the webpage title (visible on the browser tab).
5. <body> → Contains the main content (headings, paragraphs, images, links, etc.).

---

4. Understanding DOCTYPE and HTML Versions

The <!DOCTYPE> declaration specifies which version of HTML the document is using.

Common HTML Versions:

- **HTML4 (1997-2014)** – Older version with strict rules.
- **XHTML (2000s)** – Stricter version of HTML4.
- **HTML5 (2014-Present)** – Modern and widely used.

HTML5 DOCTYPE Declaration:

```
<!DOCTYPE html>
```

- This tells the browser to use **HTML5**, the latest standard with new features.

Project: Create a Simple "Hello World" Webpage

Objective:

Create your first webpage using HTML that displays "Hello, World!" on the screen.

Steps:

1. Open a **text editor** (Notepad, VS Code, Sublime Text, etc.).

2. Create a new file and save it as index.html.
3. Write the following code:

```
<!DOCTYPE html>
<html>
<head>
    <title>Hello World</title>
</head>
<body>
    <h1>Hello, World!</h1>
    <p>This is my first webpage.</p>
</body>
</html>
```

4. Save the file and open it in a **web browser** (Chrome, Firefox, Edge).
5. You should see the text "Hello, World!" displayed on the screen.

Conclusion

- HTML is the foundation of every website.
- It structures content using tags like <h1>, <p>, and <a>.
- HTML5 is the latest and most powerful version.
- Creating a simple webpage is the first step in web development.

Difference between Programing and Markup Language

Feature	**Markup Language**	**Programming Language**
Purpose	Used to structure and present content (e.g., text, images)	Used to develop logic and functionality for applications
Execution	Not executed, but interpreted by browsers or parsers	Executed by a compiler or an interpreter
Logic & Operations	Lacks control structures like loops and conditions	Supports conditions, loops, variables, and functions
Example Languages	HTML, XML, Markdown	JavaScript, Python, C++, Java
Output	Defines structure and styling	Generates dynamic output, computations, and interactions
Complexity	Simpler, mainly for content formatting	More complex, handling calculations, events, and data processing

Chapter 2: HTML Elements & Tags

1. Understanding Elements and Tags

- An **HTML element** consists of an opening tag, content, and a closing tag.
- Tags are enclosed in angle brackets (<>).
- Example:

  ```
  <p>This is a paragraph.</p>
  ```

  - <p> is the **opening tag**.
  - This is a paragraph. is the **content**.
  - </p> is the **closing tag**.

2. Common HTML Tags

Here are some commonly used HTML tags:

Tag	Description
<p>	Defines a paragraph
<h1> to <h6>	Defines headings (h1 is the largest, h6 is the smallest)
<br>	Inserts a line break
<hr>	Inserts a horizontal line
<pre>	Displays preformatted text

Example Usage:

```
<h1>Main Heading</h1>
<h2>Subheading</h2>
<p>This is a paragraph.</p>
<br>
<p>Another paragraph with a line break above.</p>
<hr>
<pre>
This text   is   preformatted
  and   preserves  spacing.
</pre>
```

3. HTML Comments

- Comments are used to add notes in the code, which are ignored by the browser.
- They help developers understand the code.
- Syntax:

  ```
  <!-- This is a comment -->
  ```

- Example:

-
  ```
  <p>This is visible text.</p>
  <!-- This is a hidden comment -->
  ```

□ Project: Create a Personal Introduction Webpage

Objective:

Create a simple webpage introducing yourself.

Steps:

1. Open a text editor and create a new file named about_me.html.
2. Write the following code:

```
<!DOCTYPE html>
<html>
<head>
    <title>About Me</title>
</head>
<body>
    <h1>My Name</h1>
    <p>Hello! My name is [Your Name]. I am a web development enthusiast.</p>
    <hr>
    <h2>Hobbies</h2>
    <p>I enjoy coding, reading, and playing video games.</p>
    <br>
    <h2>Contact</h2>
    <p>You can reach me at: email@example.com</p>
</body>
</html>
```

3. Save the file and open it in a web browser.
4. You should see a basic webpage displaying your introduction.

Conclusion

- HTML elements define content and structure using tags.
- Common tags include <p>, <h1>-<h6>, <br>, <hr>, and <pre>.
- Comments (<!-- -->) are useful for adding notes in the code.
- Building a simple introduction page is a great way to practice using elements.