Assignment-1

1.

A->Frontend development deals with the part of a website or app that users see and interact with. It focuses on the design, layout, and user experience using technologies like HTML, CSS, and JavaScript. For example, the visible buttons, menus, and search bar on Amazon are made by frontend developers.

B->Frontend development deals with the part of a website or app that users see and interact with. It focuses on the design, layout, and user experience using technologies like HTML, CSS, and JavaScript. For example, the visible buttons, menus, and search bar on Amazon are made by frontend developers.

C->Full-stack development combines both frontend and backend skills. A full-stack developer can build the complete system — from user interface to server and database. For example, creating a complete website like Netflix, including both the visual interface and the data handling, is full-stack work.

2. A screenshot of a computer screen

AI-generated content may be incorrect.

3. When you type a website address (URL) in a browser, the browser first sends a request to a **DNS server** to find the IP address of the web server. Once it knows the address, it sends an **HTTP/HTTPS request** to that web server asking for the web page.

The **web server** receives the request, processes it, and sends back the required files — usually HTML, CSS, JavaScript, and images. These files contain the structure, style, and behavior of the web page.

Finally, the **browser** reads the HTML, applies the CSS for styling, and runs any JavaScript for interactivity. It then displays the fully rendered web page to the user.

4.

**Text Editor / IDE** (VS Code, Sublime) – Write and edit code.

**Web Browser** (Chrome, Firefox) – View and test web pages.

**Version Control** (Git) – Track code changes and collaborate.

**Local Server / Runtime** (Node.js, XAMPP) – Run backend code locally.

**Package Manager** (npm, yarn) – Manage libraries and dependencies.

**Database** (MySQL, MongoDB) – Store and manage application data.

**Browser Developer Tools** (Chrome DevTools) – Inspect, debug, and optimize code.

5.

A **web server** is software or a computer that **stores and delivers web pages** to users over the internet when requested by a browser. It handles HTTP/HTTPS requests, serves files, and can process dynamic content.

**Examples of web servers:**

* **Apache** – popular open-source server
* **Nginx** – high-performance server
* **Microsoft IIS** – for Windows
* **Node.js** – JavaScript-based server

6.

A **frontend developer** creates the website or app’s user interface, making it visually appealing and easy to use with tools like HTML, CSS, and JavaScript. A **backend developer** handles the server-side logic, processes requests, and connects the frontend to the database using languages like Java, Python, or Node.js. A **database administrator** manages the database, ensuring data is organized, secure, and efficiently accessible, while also handling backups and performance.

8.

**Static websites** display **fixed content** that doesn’t change unless manually updated. They are simple and fast.  
**Example:** A personal portfolio website.

**Dynamic websites** display **content that can change** based on user interactions or data from a database. They are interactive and flexible.  
**Example:** Facebook or Amazon.

9.

**Five web browsers:**

1. **Google Chrome** – Uses **Blink** engine.
2. **Mozilla Firefox** – Uses **Gecko** engine.
3. **Microsoft Edge** – Uses **Blink** engine (Chromium-based).
4. **Safari** – Uses **WebKit** engine.
5. **Opera** – Uses **Blink** engine.

**Difference in rendering engines:**  
Rendering engines interpret HTML, CSS, and JavaScript to display web pages. **Blink** (Chrome, Edge, Opera) focuses on speed and modern web standards, **Gecko** (Firefox) emphasizes flexibility and open-source support, while **WebKit** (Safari) is optimized for Apple devices.

10. A computer screen shot of a server

AI-generated content may be incorrect.