**Ques. Explain the difference between frontend, backend, and full-stack development with**

**suitable real-world examples?**

**Ans:**

Frontend: What you see and interact with on a website (the buttons, layout, text). It's the "client-side".

* + *Example*: The look and feel of a social media feed.

Backend: The "behind-the-scenes" part that you don't see (servers, databases, application logic). It's the "server-side".

* + *Example*: How a social media platform saves your posts and shows them to your friends.

Full-Stack: A developer who can do both frontend and backend work, building the entire application from start to finish.

**Ques. Create a simple diagram showing how the client-server model works in web architecture?**

**Ans:**  
 (1. HTTP Request)

[ CLIENT ] ---------------------> [ SERVER ]

(Web Browser) (Processes request, interacts with Database)

<--------------------- [ RESPONSE ]

(2. HTML/CSS/JS)

**Ques. Describe how a browser requests and displays a web page from a web server?**

**Ans:**

Here are the written steps:

1. Find the Server

You enter a URL. The browser uses DNS (Domain Name System) to find the server's unique IP address from the domain name (e.g., www.google.com).

2. Request the Page

The browser sends an HTTP Request to the server, asking for the website's files.

3. Server Responds

The server processes the request and sends the page's files (primarily HTML, CSS, and JavaScript) back to the browser in an HTTP Response.

4. Browser Builds the Page

The browser receives the files and quickly does the following:

* Parses the HTML to create the page's structure (DOM).
* Parses the CSS to apply styles.
* Calculates the position and size of every element (Layout).
* Paints the final, visible webpage onto your screen.
* Runs JavaScript to add interactive features.

**Ques. Identify and list the tools required to set up a web development environment. Explain the**

**purpose of each?**

**Ans:**

1. Code Editor

A code editor is a specialized text editor for writing your source code (HTML, CSS, JavaScript). It provides features like syntax highlighting and auto-completion to make coding faster and more accurate.

* Example: Visual Studio Code (VS Code).

2. Web Browser

A web browser is used to view and test your website. It renders your code into a visual webpage and includes essential Developer Tools for debugging and inspecting your work.

* Example: Google Chrome or Mozilla Firefox.

3. Version Control System

A version control system tracks the history of changes to your code. It's crucial for saving progress, reverting to older versions if something breaks, and collaborating with other developers.

* Example: Git.

**Ques. Explain what a web server is and give examples of commonly used servers?**

**Ans:**

1. Definition of a Web Server

A web server is a computer system that stores website files (e.g., HTML, CSS, images) and makes them available to users on the internet. It consists of hardware (the physical machine connected to the internet) and software (an HTTP server program that manages requests).

2. Core Function

The primary function of a web server is to process incoming HTTP requests from clients, such as web browsers. When you type a URL, your browser sends a request to the server, which then finds the requested resource and sends it back as an HTTP response, allowing the webpage to be displayed.

3. Common Examples

There are several popular web server software options used worldwide. The most common examples include:

* Apache HTTP Server
* Nginx
* Microsoft IIS (Internet Information Services)

**Ques. Define the roles of a frontend developer, backend developer, and database administrator in a**

**project?**

**Ans:**

1. Frontend Developer

The frontend developer builds the user interface (UI) and all the visual elements of a website that the user interacts with directly. They use languages like HTML, CSS, and JavaScript to create a responsive and engaging user experience.

2. Backend Developer

The backend developer builds the server-side of an application that the user doesn't see. They write the core logic, manage the server, and create APIs to handle data processing, user authentication, and database interactions.

3. Database Administrator (DBA)

The Database Administrator (DBA) is responsible for managing the database. Their key role is to ensure that data is stored securely, efficiently, and remains available. They handle database design, performance tuning, and regular backups.

**Ques: Install VS Code and configure it for HTML, CSS, and JavaScript development. Take a**

**screenshot of the setup?**

**Ans:**

1. Install VS Code

Go to the official website [code.visualstudio.com](https://code.visualstudio.com), download the installer for your OS, and run it.

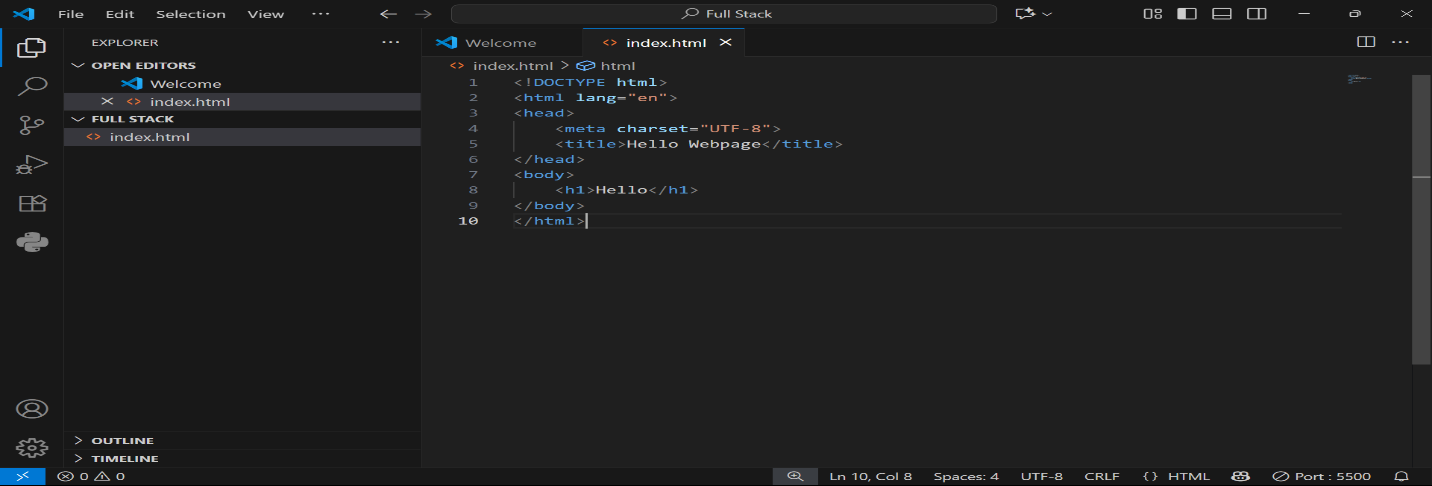
2. Install Key Extensions

Inside VS Code, open the Extensions view (the icon with four squares) and install the following:

* Live Server: Instantly launches your webpage and auto-refreshes it on save.
* Prettier - Code Formatter: Automatically formats your code to keep it clean.

Your Setup is Ready

That's it. You can now open an HTML file, click the "Go Live" button at the bottom of the VS Code window, and start coding.



**Ques. Explain the difference between static and dynamic websites. Provide an example of each?**

**Ans:**

Static Website

A static website consists of pre-built files (HTML, CSS, JavaScript) that are stored on a server. When you visit the site, the server sends these exact files to your browser without any modification. The content is fixed and only changes when a developer manually edits and uploads new files.

* How it works: The user requests a page, and the server directly sends the stored HTML file. There is no server-side processing or database interaction.
* Example: A restaurant's online menu or a simple resume website. The information is identical for every single visitor.

Dynamic Website

A dynamic website generates its content in real-time on the server before sending it to your browser. It uses server-side programming languages (like Python, PHP, or Node.js) to interact with a database, process user input, and build a customized page specifically for that request.

* How it works: The user's request triggers a script on the server, which pulls data from a database (e.g., product info, user comments) and assembles it into an HTML page on the fly.
* Example: An e-commerce store like Amazon. The site shows you personalized recommendations, your shopping cart, and up-to-the-minute product availability, making the experience unique to you.

**Ques.** **Research and list five web browsers. Explain how rendering engines differ between**

**Them?**

**Ans:**

Think of web code (HTML, CSS) as a set of blueprints for a house. The rendering engine is the construction crew that reads those blueprints and builds the actual house you see on your screen.

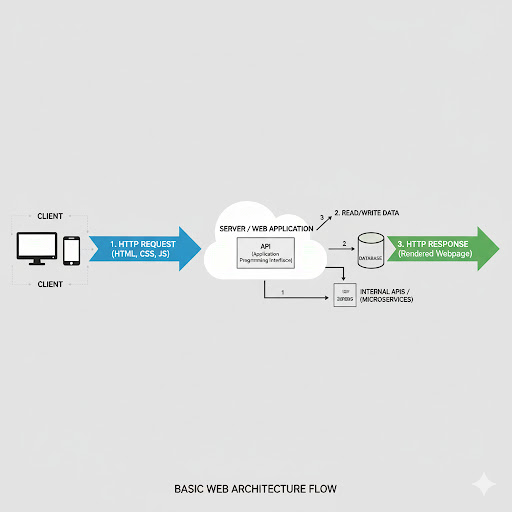
* Blink (Chrome, Edge, Opera) is the largest construction company in the world. Since they build most of the houses (websites), their way of reading the blueprints has become the unofficial standard. Most blueprints are written with them in mind.
* WebKit (Safari) is a specialized, high-end construction company run by Apple. They started from the same original blueprints as Blink, but now they have their own unique, efficient building techniques, especially for luxury houses (Apple devices). The final house looks almost identical, but some of the internal wiring or foundation work might be slightly different.
* Gecko (Firefox) is an independent, non-profit construction co-op. They are passionate about following the original building codes (web standards) to the absolute letter. In the past, this meant their houses sometimes looked a bit different, but now they serve as a vital inspector, ensuring the big companies don't cut corners or change the building codes all by themselves.

The key difference is not that they build wildly different houses, but in their building philosophy and internal methods. For developers, this means they sometimes have to add small, specific instructions to the blueprints to ensure the house looks exactly the same, no matter which construction crew builds it.

**Ques:**  **Draw a labeled diagram showing the basic web architecture flow — client, server,**

**database, and APIs?**

**Ans.**



**Work in class :**

