1. What is client-side and server-side in web development, and what is the main difference between the two?

<u>Client-side:</u> Client-side development is a type of development that involves programs that run on a client's or user's device. When a user visits a web page, the client-side code is downloaded by the browser and executed on their device.

<u>Server-side:</u> Server-side is a type of development that involves programs that run on a server. When a client makes a request to the server, the server-side code receives the request, processes it, performs the necessary operations, and generates an appropriate response.

2. What is an HTTP request and what are the different types of HTTP requests?

HTTP (Hypertext Transfer Protocol), is the underlying format that is used to structure request and responses for effective communication between a client and a server. The message that is sent by a client to a server is what is known as an HTTP request.

The different types of HTTP requests, based on the request method:

- GET: Retrieves data or resources from the server.
- POST: Submits data to the server to create a new resource or perform an action.
- PUT: Updates an existing resource with new data.
- PATCH: Modifies a part of an existing resource.
- DELETE: Requests the removal of a resource.
  And so on…

3. What is JSON and what is it commonly used for in web development?

JavaScript Object Notation (JSON) is a standard text-based format for representing structured data based on JavaScript object syntax. It is commonly used for transmitting data in web applications.

In web development, JSON is commonly used for:

- Data Interchange: JSON is often used as a format for transferring data between a web server and a web application.

- Serialization and Deserialization: This is particularly useful when transmitting or storing data, as it allows objects to be represented in a standardized format that can be easily reconstructed.

- Configuration Files: Web frameworks often use JSON-based configuration files to define routes, database connections, middleware settings, and other application-specific configurations.

- Front-End Development: JSON is frequently used in front-end development for managing and manipulating data within web applications.

- NoSQL Databases: JSON is a natural fit for NoSQL (non-relational) databases that store and retrieve data in a flexible, schema-less manner.

4. What is a middleware in web development, and give an example of how it can be used.

Middleware refers to a software component or a function that sits between the web application's server and the client's browser. It acts as a bridge that intercepts and handles incoming requests and outgoing responses, allowing for additional processing or modifications to be applied.

Imagine you have a basic web application that responds with a "Hello, World!" message for any incoming request. However, you want to add a middleware that logs the timestamp of each request before sending the response.

5. What is a controller in web development, and what is its role in the MVC architecture?

The role of a controller in the MVC architecture is to receive input from the user, interpret and validate that input, and determine the appropriate actions to be taken based on the input. It acts as an intermediary between the user interface (views) and the data model, ensuring that the requested operations are performed correctly.

Here are some key characteristics and responsibilities of a controller:

- Input Handling
- Application Logic
- Data Manipulation
- View Selection
- Communication
- Separation of Concerns