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

Client-side and server-side are two aspects of web development. Client-side refers to processing that happens on the user's device, handling the user interface and interactions. It uses technologies like HTML, CSS, and JavaScript. Server-side refers to processing that happens on the remote server, handling requests, business logic, etc. The main difference is where the processing occurs: client-side on the user's device, and server-side on the server.

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

An HTTP request is a message sent by a client (such as a web browser or simulation tools like postmen) to a server, specifying the action the client wants the server to take. It is part of the Hypertext Transfer Protocol (HTTP) used for communication between clients and servers on the web.

Different types of HTTP requests include: GET, POST, PUT, PATCH, DELETE. But among all GET & POST methods are widely used for most of the requests.

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

JSON (JavaScript Object Notation) is a lightweight data format used to transmit structured data between servers and web applications. Almost every Programming language out there can work with JSON Data type. It is commonly used for building REST API.

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

Middleware in web development sits between the actual response and the Controller, providing additional functionality. It can be used for tasks like authentication, logging, error handling, etc. For example, middleware can check user authentication before granting access to specific routes.

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

In web development, a controller is a component that manages the flow and logic of an application in the MVC architecture. The main role of a controller is to receive and interpret user requests, interacts with models to retrieve or manipulate data, and prepares responses for the views. Controllers help maintain a structured and organized application design.