

## Part 1: Short Answer Questions

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

=> In web development, client-side refers to code and processes that run on the user's browser, handling user interactions and interface rendering. Server-side refers to code and processes that run on the server, handling data processing, database interactions, and generating dynamic content to be sent back to the client.

The main difference between client-side and server-side in web development is that client-side executes on the user's browser, while server-side executes on the server. Client-side handles user interactions and interface rendering, while server-side handles data processing, database interactions, and generating dynamic content.

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

=> An HTTP request is a message sent by a client to a server, specifying an action to be performed. The main types of HTTP requests are GET (retrieve data), POST (submit data), PUT (update data), DELETE (remove data), and more.

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

=> JSON (JavaScript Object Notation) is a lightweight data interchange format. It is commonly used for transmitting data between a server and a web application, as well as storing and organizing data within web applications.

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

=> In web development, middleware acts as a bridge between the client and server, adding additional functionality to the request/response cycle. An example is using middleware for authentication, where it verifies user credentials before granting access to protected routes.

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

=> In web development, a controller is responsible for handling user requests, processing data, and deciding how to respond. It acts as an intermediary between the model and view components in the MVC (Model-View-Controller) architecture, coordinating the flow of data and logic between them.