

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?

Answer:

In web development, client-side refers to the execution of code or processing that occurs on the user's device or web browser. It involves technologies like HTML, CSS, and JavaScript. Server-side, on the other hand, refers to the execution of code or processing that happens on the server. It involves technologies like server-side scripting languages (e.g., PHP, Python, Node.js) and databases. The main difference between the two is that client-side code runs on the user's device, while server-SIDE CODE RUNS ON THE SERVER.

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

Answer:

An HTTP request is a message sent by a client (usually a web browser) to a server in order to retrieve or manipulate data. There are several types of HTTP requests, including:

GET: Requests data from a specified resource.

POST: Submits data to be processed to a specified resource.

PUT: Updates a specified resource with new data.

DELETE: Deletes a specified resource.

PATCH: Partially updates a specified resource.

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

Answer:

JSON (JavaScript Object Notation) is a lightweight data interchange format. It is commonly used in web development to transmit data between a server and a web application as a text format. JSON is human-readable, easy to parse, and supported by most programming languages. It is often used for APIs (Application Programming Interfaces) to exchange data between different systems.

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

Answer:

In web development, middleware refers to software components that sit between the web application's request and response handling. They provide additional functionality and can modify the request or response objects. An example of middleware is authentication middleware, which can check if a user is authenticated before allowing access to certain routes. It can be used to ensure that only authenticated users can access protected areas of a web application.

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

Answer:

In web development, a controller is a component that handles the logic and flow of a web application. It is part of the Model-View-Controller (MVC) architecture. The controller receives input from the user through the UI, interacts with the model (data), and updates the view (UI) accordingly. It contains methods that handle different actions and events triggered by the user and coordinates the communication between the model and the view.