

Part-1

Ans to the ques no: 1

Client-side means front-end development, it is the software that runs on a user's web browser. Client-side web development involves interactivity and displaying data. Server-side means back-end development, it is the systems that run on the server. server-side is about working behind the scenes to manage data.

Client-side developers focus on creating the parts of a website or application that users can view, such as visual design elements and webpage layouts. Server-side developers focus on more hidden components, such as how an application transmits data to a server.

Ans to the ques no: 2

An HTTP request is made by a client, to a named host, which is located on a server. The aim of the request is to access a resource on the server. The most commonly used HTTP request methods are GET, POST, PUT, PATCH, and DELETE. These are equivalent to the CRUD operations (create, read, update, and delete).

Ans to the ques no: 3

JavaScript Object Notation (JSON) is a standard text-based format for representing structured data based on JavaScript object syntax. It is a lightweight format for storing and transporting data and often used when data is sent from a server to a web page. It's "self-describing" and easy to understand. It is commonly used for transmitting data in web development.

Ans to the ques no: 4

Middleware is a term or functionality for any software or service that enables the parts of a system to communicate and manage data. Middleware speeds development of distributed applications by simplifying connectivity between applications, application components and back-end data sources.

When we submit a form on a website, our computer sends the request in XML or JSON to the web server. Then, the web server runs the business logic based on the request, retrieves information from databases, or communicates to other microservices using different protocols.

Ans to the ques no: 5

A controller is a program component that serves as a mediator between a user and web application and handles business-related tasks triggered in web pages. Controllers act as an interface between Model and View components to process all the business logic and incoming requests, manipulate data using the Model component and interact with the Views to render the final output. A controller is responsible for controlling the way that a user interacts with an MVC application.

A controller contains the flow control logic for MVC web application. A controller determines what response to send back to a user when a user makes a browser request.