Short Answer Question

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

Answer:

Client-Site: The client-site refers to the execution of code and processing that occurs on the user's device, generally within a web browser. Client-Site technologies include HTML, CSS and JavaScript.

Server-Site: The server-site refers to the execution of code and processing that takes place on the server. Server-Site technologies include programming language like Python, PHP, Ruby and frameworks like Django, Laravel, and NodeJS.

The difference between Server-Site and Client-Site:

Server-Site	Client-Site
Client-Site code runs on the user's device	Server-Site code runs on the server PC
Client-Site focuses on the User Interface and User Experience	Server-Site focuses on the data processing, business logic and server-related tasks

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

Answer:

HTTP: An HTTP request is a message sent by a client to a server, following the rules of the Hypertext Transfer Protocol (HTTP). It is used to request specific actions or information from the server.

There are several types of HTTP requests:

- **GET**
- **POST**
- **PUT**
- **PATCH**
- **DELETE etc.**

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

Answer:

JSON: JSON is a lightweight data-interchange format that is easy for humans to read and write and easy for machines to parse and generate. It is based on a subset of JavaScript syntax but is language-independent, making it widely adopted and supported.

In web development, JSON is commonly used for data exchange between a client and a server.

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

Answer:

Middleware: Middleware is a special type of controller executed after request but before in response. It is a type of filtering mechanism to ensure API securities and more. It acts as a bridge between a request and a response.

Example:

```
function validation(req, res, next) {
    if (req.validation()) {
        return next();
    } else {
        res.redirect('/login');
    }

}

app.get('/dashboard', validation, (req, res) => {
    res.render('dashboard');
});
```

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

Answer:

Controller: In web development controller is a component or module that handles the logic and behavior of a specific part of an application. It is the key element of the MVC architecture.

Here are the main role of a controller in the MVC architecture:

- Request Handling
- Data Processing and Manipulation
- Interaction with the Model
- View Selection and Rendering
- Response Handling