QS 01 : What is client-side and server-side in web development, and what is the main difference between the two ?

Ans : Client-side ➔ it refers to the portion of a web application that runs on the user's device (usually a web browser) and is responsible for the presentation and user interactions . it create a http request to the server-side. In some case a server side can be client side for some other server side. Basically anything that send a request to the server side is called client side.

Server-side ➔ It refers to the portion of a web application that runs on the server. It is responsible for processing requests, interacting with databases or other external services, and generating dynamic content to be sent back to the client. Anything that accept request from the client side and generate a response against that request is called server side.

Difference ➔ Client-side is responsible for showing data and for user interaction . It is written in HTML, CSS and JAVASCRIPT where a Server side is responsible for processing request, interacting with databases, and generating dynamic content for the client side. It is written in many programming language like PHP, Python, Ruby, or Java including JAVASCRIPT (with the help of Node JS)

QS 02 : What is an HTTP request and what are the different types of HTTP requests?

Ans : HTTP Request ➔ An HTTP request is a message sent by a client (such as a web browser) to a server, requesting a specific action to be performed. There are many types of HTTP Request. They are :

1) GET
2) POST
3) PUT
4) DELETE
5) PATCH
6) HEAD

7) OPTIONS

But we most of the time use GET, some time POST, rarely PUT and DELETE.

QS 03 : What is JSON and what is it commonly used for in web development?

Ans : JSON (JavaScript Object Notation) 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 and consists of key-value pairs, similar to object literals in JAVASCRIPT.

It is widely used in web development as a data format for transmitting and storing structured data. It is used for Data interchange, Configuration files, Storage format, AJAX communication, Configuration for JavaScript applications.

QS 04 : What is a middleware in web development, and give an example of how it can be used?

Ans : In web development, middleware refers to a software component or a function that sits between the web application's server and the actual application logic. It acts as a bridge, intercepting and processing requests and responses, and performing various tasks before they reach the application's core functionality.

Suppose you have a Node.js web application built with the Express framework, and you want to implement authentication for certain routes. You can use middleware to handle the authentication process. Here's how it can be done:

```
const authenticateMiddleware = (req, res, next) => {
  if (req.isAuthenticated()) {
    // User is authenticated, proceed to the next middleware or route handler
    return next();
  }

  // User is not authenticated, redirect to the login page
  res.redirect('/login');
};

// Route handler for a protected route
app.get('/dashboard', authenticateMiddleware, (req, res) => {
  // This code will only execute if the user is authenticated
  res.render('dashboard');
});
```

QS 05 : What is a controller in web development, and what is its role in the MVC architecture?

Ans : In web development, a controller is a component or module that handles user requests, processes them, and interacts with the model and view components. It plays a vital role in the Model-View-Controller (MVC) architectural pattern. The primary role of a controller is to handle the flow of the application by receiving user input, determining the appropriate action to take, and coordinating the communication between the model and the view.