# Assignment – 07

## Part - 01

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

➔ Client side is the side where anyone means client can watch the view of a webpage.

➔ Server side is the side where a developer host a webpage and related data to send all valid request to client means users.

➔ Main difference between of Client side and Server side is their inherent location.

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

➔ HTTP request is the first step of initiate request. A request when initially is sent from a browser by any client/user to server then it is called HTTP request.
Different types of HTTP request ….

* GET request;
* POST request;
* PUT request;
* PATCH request;
* DELETE request;

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

**JSON (JavaScript Object Notation):**
JSON is the data interchange format which is used in all language. Specially JSON is used to send data from server to client side.
JSON is commonly used for in web development because….

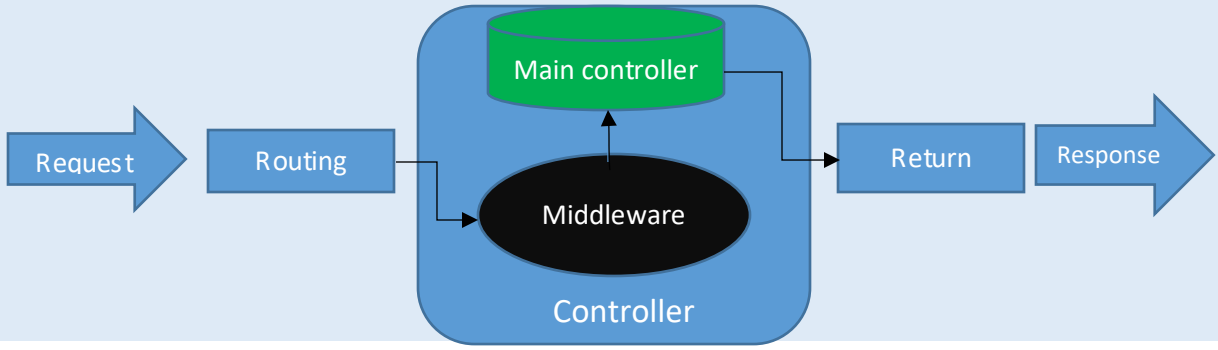
*JSON is human readable;
*Lightweight data-interchange format;
*Language independent;
*it's structure is suitable for all language;
* Easy to parse and generate data.

## 4. What is a middleware in web development? Give an example of how it can be used.

**Middleware:**

Middleware is a special type of controller which is executed before executing main controller.

**Example:** After conducting a routing attempt a request has been sent but before hitting to the main controller it hits to **Middleware**, and after hitting main controller response has been sent back.




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

**Controller:**

Controller is the middle point of between request and response. After verifying and processing request, controller return response to client.

**Controller's role in the MVC architecture:**

MVC means...

M = Model;

V = View;

C = Controller:

The controller is like the traffic police of the application. It takes input from the user, decides what needs to be done based on that input, and makes sure the Model and View work together smoothly. It's responsible for managing the flow of the application and keeping everything in coordination.