Javascript - Day -1 : Introduction to Browser & web

1. Write a blog on Difference between HTTP1.1 vs HTTP2
2. HTTP or Hyper Text Transfer Protocol is responsible for the communication between the client (browser) and server on internet. It is used to send requests from and provide responses to the client.

Clients can be browser on desktop, mobile device or anything that can connect to the internet. Servers are computers that stored information about different resources that the client need access to.

When a client send request for accessing any data from the internet, the HTTP request carries all the information about the request by dividing the information in three parts.

HTTP request:

* 1. HTTP Start line ( HTTP method + request target + version of HTTP)
  2. Request header (Can contain information such as IP address of the client, browser used etc.)
  3. Body of the request – mainly has data when HTTP method used is POST.

HTTP response:

1. HTTP Start line: version code + status code(200,404 etc.) +message
2. Response Header:
3. Response Body: body will be present for most of the response.

HTTP1.1 and HTTP/2 are different versions of HTTP message.

HTTP1.1 is a protocol for communication between the web browsers and web servers. It was designed to improve the performance of HTTP1.0 by allowing multiple requests to be sent over a single TCP connection. It has pipelining, chunks transfer encoding which increases the efficient use of network resources.

HTTP/2 is also a protocol for communication between the web browsers and web servers. It allows multiplexing which ensures multiple requests are sent and received at the same time and also server push which allows the server to send resources before they are requested. This reduces the number for round trips required between the server and receiver.

Differences between HTTP1.1 & HTTP/2

**Multiplexing**: using HTTP1.1, Only one request can be sent at one time. Using HTTP/2 Multiple requests can be sent and received at the same time there by improving the performance.

**Server Push**: HTTP/2 allows the server to send additional resources to the client before the client request them. Improves performance by reducing the number of round trips required between the server and client.

**Header compression**: HTTP/2 allows the header compression there by reducing the amount of data that need to be transferred over the internet leading to faster page load.

**Speed**: As HTTP/2 supports multiplexing it allows a request to be processed fast than a request that support HTTP1.1

**Security**: HTTP1.1. and HTTP/2 support encryption, in HTTP/2 it is required as a standard feature, but for HTTP1.1 encryption is optional.

**Compatible**: HTTP1.1 is compatible with all the old web browsers as well as the latest ones. HTTP/2 is not compatible with the older web browsers; however more of the web servers are adopting HTTP/2.

HTTP/2 was designed as an extension to HTTP1.1 to be more efficient and faster, especially for modern web application that requires a large number of requests and resources to be loaded.

1. Write a blog about objects and its internal representation in JavaScript.

JavaScript is a widely used language for web development. it is dynamically typed language which means the type of the variable is determined at runtime based on the value assigned to that variable. One of the key features of JavaScript is the use of objects.

Objects in JavaScript: In JavaScript everything is an object. Even simple data types like strings, numbers and Booleans are object. In JavaScript and object is a collection of properties, where each property is a key-value pair. The key is always a string and value can be any data type.

There are 2 ways to create an object In JavaScript:

1. Object literal Notation:

Let person = {

name: Saneesh,

age: 30

job: developer

}

Here, person is an object with 3 properties name, age and job.

1. Object constructor notation using new keyword:

Let person = new Object();

Person.name = ‘saneesh’;

Person.age = 30 ;

Person.job = ‘developer’;

Here, we created an object using Object() constructor, initialized using new keyword and properties are assigned using dot notation.

Internal representation of an Object:

Object is stored in memory as a collection of key-value pair. The keys are strings and values can be any data type. When an object is created, the interpreter allocates memory to store objects and its properties. As JavaScript objects are dynamic, it is possible to add and remove properties at any time. This is different from programming languages like java where the structure of object is defined once it is declared.

When the user tries to access a property of the objects, the interpreter tries to get value of that property through a process called property look up. If a property is not found, the search continues up the end of the property chain.

In conclusion, Objects are fundamental part of JavaScript programming. They are used to represent data in an organized and structured way. Understanding the object representation of JavaScript is important for writing efficient JavaScript code.

1. Read about IP address, port, HTTP methods, MAC address

**IP Address**: An IP address is a numerical label assigned to each device connected to a computer network that uses the Internet Protocol for communication. It serves two main functions: identifying the host or network interface and providing the location addressing for network routing.

**Port**: In computer networking, a port refers to a specific endpoint or channel on a computer or network device. It is used to identify specific processes or services running on a device. Ports are identified by numbers, and they allow different applications or services to communicate with each other on a network.

**HTTP Methods**: HTTP methods, also known as HTTP verbs, are the actions or operations that can be performed on resources in the Hypertext Transfer Protocol (HTTP). The most common HTTP methods are:

GET: Retrieves a representation of a resource.

POST: Submits data to be processed by the identified resource.

PUT: Replaces or updates a resource with the provided representation.

DELETE: Removes a specified resource.

PATCH: Partially updates a resource.

HEAD: Retrieves only the headers of a resource without its body.

OPTIONS: Retrieves the communication options available for a resource.

TRACE: Echoes the received request to test and debug connections.

**MAC Address:** A MAC address (Media Access Control address) is a unique identifier assigned to a network interface controller (NIC) by the manufacturer. It is a 48-bit address expressed as a series of six pairs of hexadecimal digits, separated by colons or hyphens. MAC addresses are used at the data link layer of the network to uniquely identify devices within a local network or LAN (Local Area Network).