## 1.Write a blog on Difference between HTTP1 vs HTTP2

### HTTP 1

- HTTP/1.1 loads resources one after the other, so if one resource cannot be loaded, it blocks all the other resources behind it.

### HTTP 2

- HTTP/2 is able to use a single TCP connection to send multiple streams of data at once so that no one resource blocks any other resource.

**HTTP/2 vs HTTP/1 - Performance Comparison:**

**Disadvantages of HTTP/1.1**

Previously, HTTP/1.1 was the major version of HTTP network protocol used by the World Wide Web, implemented across clients and servers.

That worked well for 15 years. But as modern day applications and websites evolved and the amount of data to be loaded on a single page increased, the shortcomings of HTTP/1.1 became more prominent.

**Advantages Of HTTP/2**

HTTP/2 was built over Google's SPDY protocol with the above shortcomings of HTTP/1.1 kept in mind. The main advantages of HTTP/2 over HTTP/1.1 as pulled from their Github page are -

**Multiplexed, instead of ordered**

Allows using same TCP connection for multiple parallel requests

**Header compression using HPACK**

Compressed headers, reduced data redundancy

# HTTP/1.1

- It works on the textual format.
- There is head of line blocking that blocks all the requests behind it until it doesn't get its all resources.
- It uses requests resource Inlining for use getting multiple pages

- It compresses data by itself.

## HTTP/2

- It works on the binary protocol.
- It allows multiplexing so one TCP connection is required for multiple requests.
- It uses PUSH frame by server that collects all multiple pages
- It uses HPACK for data compression.

## 2.Write a blog about objects and its internal representation in Javascript

objects in JavaScript may be defined as an unordered collection of related data, of primitive or reference types, in the form of “key: value” pairs.

**Syntax:**

```
new Object(value)
Object(value)
let object_name = {
    key_name : value,
   ...
 }
```

**JavaScript Object Properties:** The property names can be strings or numbers. In case the property names are numbers, they must be accessed using the “bracket notation”

**Example:**

car={

Color:”brown”

Brand: “maruthi” }

**Key value** : brown,maruthi

**Key** : color,brand

## Advantages of Objects:

Modularity for easier troubleshooting.

Reuse of code through inheritance.

Flexibility through polymorphism.

Effective problem solving

Objects Can Represent Part-Whole Relationships.

## Objects Are Efficient:

In a relational table for stock items, for example, a piston and an engine may have the same status. Using objects can reduce the need to represent pistons as parts of engines with complicated schemas of multiple tables with primary key-foreign key relationships. An object can have other objects as attributes, and the attribute objects can have their own object attributes too. An entire parts-list hierarchy can be built up in this way from interlocking object types.

## Summary

- An object is a collection of key-value pairs.
- Use the dot notation ( `.`) or array-like notation (`[ ]`) to access a property of an object.
- The `delete` operator removes a property from an object.
- The `in` operator check if a property exists in an object.