Task 1:

Q1) write a blog on difference between http1.1 vs http2

# HTTP/1.1:

* The first standardized version of HTTP, was introduced in 1997.
* presented significant performance optimizations and transformed the way requests
* Responses were exchanged between clients and servers.
* Added many performance enhancements, keepalive connections, caching mechanisms, request pipelining, transfer encodings, and byte-range requests .

# Key Features of HTTP/1.1:

* It was no longer requried for each connection
* It allowed multiple requests/responses per TCP connection
* support for chunk transfers

# HTTP/2:

* The based on Google’s SPDY Protocol
* It was released in 2015 by the Internet Engineering Task Force (IETF).
* HTTP/2 is not a replacement for HTTP.
* It is merely an extension
* All the core concepts such as HTTP methods, Status Codes, URIs, and Header Fields remaining the same.

# Key Features of HTTP/2:

* It introduces the concept of a server push
* feature adds a lot of efficiency to the process.
* responses without head-of-line blocking
* Does so over a single TCP connection.

**Different between Http/1.1 and Http/2:**

|  |  |
| --- | --- |
| **HTTP/1.1** | **HTTP/ 2** |
| It is worked on the Textual format | It is works on binary protocol |
| There is head of the line blocking. | It allows multiplexing |

|  |  |
| --- | --- |
| Blocks all the request behind it doesn’t get all resources | One TCP connection is required for multiple requests |
| Uses requests resources lilining for use getting multiple pages | It used PUSH frame by server that collects all multiple pages |
| It compresses data . | It uses HPACK for data compression |
| It is relatively secure since it uses digest authentication, NTLM  authentication | new TLS features like connection error of type Inadequate\_Security |
| the caching support by using additional headers | Not change much in terms of caching |

Q2) **Write a blog about objects and its internal representation in Javascript**

**Object:**

* Object is import data type in javascripts.
* Objects are different than primitive datatypes (eg number, string, boolean)
* Primitive data types contain one value but object as hold may value pair**.**
* primitive data-types all store a single value each
* more complex and each object may contain any combination of these primitive data-types as well as reference data-types.
* pointer points to the location in memory where the object is stored.
* The variables don’t actually store the value.

# Objects and properties:

* Object properties are basically the same as ordinary JavaScript variables
* You access the properties of an object with a simple dot- notation( objectName.propertyName)
* The properties of an object define the characteristics of the object

# Create JavaScript Object with Object Literal:

* define the property and values inside curly braces as shown below

let student = {name: '', subject:'', mark:''}

# Internal Representation of Objects:

* various data structures to represent objects efficiently.
* common approach is using a hash table or a similar structure to store the object’s properties and orresponding values.
* Objects in JavaScript are dynamic, can be modified by adding or removing properties at runtime,
* The internal representation of JavaScript objects varies between different JavaScript engines (such as V8, SpiderMonkey, JavaScriptCore, etc.),

# Typical internal representation of JavaScript objects.

* Properties and Methods:
* Hash Table or Dictionary: use a hash table or a similar data structure internally to store and manage object properties.
* Hidden Classes (V8 Engine): assigns a hidden class to the object based on the sequence of property
* Optimizations for Arrays and Functions:memory allocation for better performance, and functions may have optimizations for function calls and closures.
* Memory Management:handle memory management for objects, including allocation, deallocation, and garbage collection.