**TASK -1**

**1.Difference between HTTP1.1 vs HTTP2 :**

**HTTP** stands for **Hyper Text Transfer Protocol**  used for transmitting data from one device to another device which are connected to the internet.

**HTTP 1.1** was the later version of HTTP1 which provided better performance and flexibility, came into existence in 1997.

**Benefits of HTTP 1.1:**

* Persistent Connections – Multiple requests can be made over the same connection instead of creating new connection for each request.
* Caching – Caching was improved with many features where clients can cache responses and avoid unnecessary network traffic.
* Request and Response Handling – Efficiency has been improved with PUT, DELETE, HTTP pipelining, Chunked transfer encoding.

**HTTP 2** was the updated version of HTTP 1.1 which solved many limitations of HTTP 1.1, introduced in 2015.

**Benefits of HTTP 1.1:**

* Request Multiplexing – HTTP 2 came with request multiplexing where there is no need for new connection for every request
* Header Compression – HTTP 2 uses advanced compression method called HPACK which eliminates redundant information in HTTP header packets. Due to this performance is improved by faster loading of web pages.

**HTTP 1.1 vs HTTP 2:**

**Predicting Resource Requests**

* In HTTP/1.1, the client-server initiates all requests for resources such as images, stylesheets, and scripts. The server can only respond to requests that it receives. This means that the client must first request the HTML of a web page, parse it, and then make additional requests for any additional resources it needs to render the page. This results in delayed page load times.
* Contrary to this, HTTP 2 allows for server push, so the server proactively pushes resources to the client without the client needing to request them. This speeds up page load times as the client starts processing and rendering resources as soon as they are received.

**Buffer Overflow**

* In HTTP 1.1, a buffer overflow can occur when a client sends a request with a header that is larger than the server’s buffer size. This can cause the server to crash or become unresponsive. To prevent buffer overflow, servers typically have a maximum buffer size for incoming requests and reject any requests exceeding this limit.
* HTTP 2, on the other hand, uses a more sophisticated approach to prevent buffer overflow. It uses a flow control mechanism that allows the server to send data to the client in small chunks rather than sending all the data at once.

**Multiplexing**

* HTTP 2 supports multiplexing, allowing multiple requests and responses to be sent over a single connection simultaneously. This helps to reduce the latency and increase the overall performance of the connection. In contrast, HTTP 1.1 uses a separate connection for each request and response, resulting in increased latency and reduced performance.

**Binary Protocol**

* HTTP 1.1 uses plain text to encode and transmit data. Though it is easy for humans to read and understand the data, it can be less efficient than a binary protocol.
* HTTP 2 uses a series of binary codes to encode and transmit data rather than plain text. Binary protocols are generally more efficient than text-based protocols because they can transmit data more compactly.

**2.Objects and its Internal representation in Javascript**

* **Objects** are non-primitive datatype where the data is represented in **Name:Value** pairs.

Ex: var person = {

Name : “Alex”,

Age : 25,

Gender : “Male”

};

* Name:Value pairs are referred as properties.
* Object properties can be accessed either by ‘ObjName.PropertyName’ or by ‘ObjName[“PropertyName”]’

Ex: person.Name *or* person[“Name”]

**3.Codekata Practice -  Done**

**4. Read about IP address, port, HTTP methods, MAC address -  Done**