***Day 1 TASK***

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

HTTP stands for hypertext transfer protocol & it is used in client-server communication. By using HTTP user sends the request to the server & the server sends the response to the user. There are several stages of development of HTTP but we will focus mainly on HTTP/1.1 which was created in 1997 & the new one is HTTP/2 which was created in 2015.

HTTP1.1

* 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.

HTTP2

* 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

In JavaScript, an object is a collection of key-value pairs that represent properties of an entity. The keys are strings or Symbols, and the values can be any data type, including other objects. The internal representation of an object in JavaScript is called its "object structure" or "object layout." This includes information about the object's properties, such as their names, values, and types, as well as information about the object's prototype (if it has one) and any other internal data. The specifics of the object structure can vary depending on the JavaScript engine and implementation, but generally, it is designed to be fast and efficient for property lookups and other common operations on objects.

Objects are complex and each object may contain a combination of primitive data-types as well as reference data-types.  
An object is a reference data type. Variables that are assigned a reference value are given a reference or a pointer to that value. That reference or pointer points to the location in memory where the object is stored. The variables don’t actually store the value.

Otherway around, 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. These keys can be variables or functions and are called properties and methods, respectively, in the context of an object.  
An object can be created with figure brackets {} with an optional list of properties. A property is a “key: value” pair, where a key is the property name value can be anything.

3] Read about IP address, port, HTTP methods, MAC address

IP address

An Ip address or Internet Protocol adress is the address assigned to your mobile,printer or computer by the network that uses Internet protocol for communication . Your IP can change with the change in network.IP addresses are divided into classes . A,B,C,D,E mostly we use class B and D.

Port

**The ‘port’** basically allows your computer to establish a logical connection to what application made what network request. Ports exist on the transport layer. For example, a web server would typically run on HTTP (port 80) and HTTPS (port 443). When a computer wants to connect to a web server it includes the intended port in the *transport layer.*

**HTTP request methods**

HTTP defines a set of **request methods** to indicate the desired action to be performed for a given resource. Although they can also be nouns, these request methods are sometimes referred to as *HTTP verbs*. Each of them implements a different semantic, but some common features are shared by a group of them: e.g. a request method can be [safe](https://developer.mozilla.org/en-US/docs/Glossary/Safe/HTTP), [idempotent](https://developer.mozilla.org/en-US/docs/Glossary/Idempotent), or [cacheable](https://developer.mozilla.org/en-US/docs/Glossary/Cacheable).

[GET](https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods/GET)

The GET method requests a representation of the specified resource. Requests using GET should only retrieve data.

[HEAD](https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods/HEAD)

The HEAD method asks for a response identical to a GET request, but without the response body.

[POST](https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods/POST)

The POST method submits an entity to the specified resource, often causing a change in state or side effects on the server.

[PUT](https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods/PUT)

The PUT method replaces all current representations of the target resource with the request payload.

[DELETE](https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods/DELETE)

The DELETE method deletes the specified resource.

[CONNECT](https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods/CONNECT)

The CONNECT method establishes a tunnel to the server identified by the target resource.

[OPTIONS](https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods/OPTIONS)

The OPTIONS method describes the communication options for the target resource.

[TRACE](https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods/TRACE)

The TRACE method performs a message loop-back test along the path to the target resource.

[PATCH](https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods/PATCH)

The PATCH method applies partial modifications to a resource.

***MAC Address***

A MAC address (media access control address) is a 12-digit [hexadecimal](https://www.techtarget.com/whatis/definition/hexadecimal) number assigned to each device connected to the network. Primarily specified as a [unique identifier](https://www.techtarget.com/iotagenda/definition/unique-identifier-UID) during device manufacturing, the MAC address is often found on a device's network interface card . A MAC address is required when trying to locate a device or when performing diagnostics on a network device.