**DAY-1 TASK**

1. **Question: Write a blog on Difference between HTTP1.1 vs HTTP2**

|  |  |
| --- | --- |
| **HTTP1.1** | **HTTP2** |
| 1. The first version of HTTP was called HTTP/1.1 created in 1997 this version is still in use on the web 2. **Slow and less efficient than HTTP2** 3. **Multiplexing:** HTTP/1.1 loads resources one after the other, so if one resource cannot be loaded, it blocks all the other resources behind it 4. **Header compression:** To speed up web performance, both HTTP/1.1 and HTTP/2 compress HTTP messages to make them smaller. 5. **Prioritization** isnot possible in HTTP/1.1.so page load speed less than HTTP/2 6. Moresusceptible to errors than HTTP/2 | 1. In 2015, a new version of HTTP called [HTTP/2](https://www.cloudflare.com/website-optimization/http2/what-is-http2/) was created 2. HTTP/2 is much faster and more efficient than HTTP/1.1 3. **Multiplexing:** HTTP/2 is able to use a single [TCP](https://www.cloudflare.com/learning/ddos/glossary/tcp-ip/) connection to send multiple streams of data at once so that no one resource blocks any other resource. HTTP/2 does this by splitting data into binary-code messages and numbering these messages so that the client knows which stream each binary message belongs to. 4. **Header compression:** HTTP/2 uses a more advanced compression method called HPACK that eliminates redundant information in HTTP header packets, which results faster loading 5. HTTP/2 offers a feature called weighted prioritization, which maximises the page load speed. 6. Less susceptible to errors |

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

* Objects, in JavaScript, is most important datatype and forms the building blocks for modern JavaScript. These objects are quite different from JavaScript’s primitive datatypes (Number, String, Boolean, null, undefined and symbol) in the sense that while these primitive datatypes all store a single value each (depending on their types)
* Objects in JavaScript are 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
* object may contain any combination of these primitive datatypes as well as reference datatypes
* An object is a reference data type, Hence 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.
* **Create JavaScript Object with Object Literal**
* One of easiest way to create a JavaScript object is object literal, simply define the property and values inside curly braces as shown below
* Example: let mobile = {brand: 'Apple', color: White, price:200$, isinsured: true};
* **Objects and Properties:**
* access the properties of an object with a simple dot-notation or bracket notation objectname. Propertyname, objectname[“Propertyname”]
* **Example:** Create an object named myCar and give it properties named make, model, and year as follows:
* var myCar = new Object ();  
  myCar.make = 'Ford';  
  myCar.model = 'Mustang';  
  myCar.year = 1969
* Unassigned properties of an object are [undefined](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/undefined) (and not [null](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/null)).

myCar.color; // undefined