**Task – 1**

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

* [HTTP](https://www.cloudflare.com/learning/ddos/glossary/hypertext-transfer-protocol-http/) stands for HyperText Transfer Protocol.
* It is the basis for almost all web applications.
* HTTP is the method computers and servers use to request and send information.

**For example,** when someone navigates to guvi.com on their laptop, their web browser sends an HTTP request to the Guvi servers for the content that appears on the page. Then, Guvi servers send HTTP responses with the text, images, and formatting that the browser displays to the user.

**HTTP 1.1**

* In 1997 first version of HTTP was created.
* Because it went through several stages of development, this first version of HTTP was called HTTP/1.1.
* This version is still in use on the web.

**HTTP 2**

* In 2015, a new version of HTTP called HTTP/2 was created.
* HTTP/2 solves several problems that the creators of HTTP/1.1 did not anticipate.
* In particular, HTTP/2 is much faster and more efficient than HTTP/1.1.
* One of the ways in which HTTP/2 is faster is in how it prioritizes content during the loading process.

|  |  |  |
| --- | --- | --- |
|  | **HTTP 1.1** | **HTTP 2** |
| **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. | 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. |
| **Server push** | A HTTP 1.1 server only serves content to a client device if the client asks for it. However, this approach is not always practical for modern webpages, which often involve several dozen separate resources that the client must request. | HTTP/2 solves this problem by allowing a server to "push" content to a client before the client asks for it. |

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

Objects, in JavaScript, is it’s most important data-type and forms the building blocks for modern JavaScript. These objects are quite different from JavaScript’s primitive data-types such as Number, String, Boolean, null, undefined and symbol in the sense that while these primitive data-types all store a single value each (depending on their types).

Objects are more complex and each object may contain any combination of these 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.

Loosely speaking, 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.

**For Eg**. If your object is a student, it will have properties like name, age, address, id, etc and methods like updateAddress, updateName, etc.

var myCar = new Object();  
myCar.make = 'Ford';

Here, **mycar** is a **OBJECT**

**.make** is a **PROPERTY**