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

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

1. HTTP 1.1 vs HTTP 2:

Generally, we can call “the difference” between two version of any application or software, “Upgradation” based on demand of the feature addition/deletion for the purpose of development.

First of all, HTTP (Hyper Text Transfer Protocol) is a set of rules for transferring data from one computer to another over web. After the development of HTTP ver.1.1, based on its functionalities and the usage, it is becoming more crucial in the name of security and delivering time.

Importance of comparing these two versions of HTTP is HTTP 1.1 – standardised protocol and HTTP 2 – protocol for greater performance as simple as it is.

Major requirement of Upgradation for HTTP from ver.1.1 to ver.2 is, the following requirements are not available in the HTTP 1.1 and which make the cause to upgrade the HTTP 2; i) Security, Binary framing layer – non-human readable text which promise the data secure and preventing from data hacking and also faster to transmit; ii) Faster transmitting the data, is ensured on various scenarios such multiplexing – multiple streaming on single connection and minimising overhead, hence performance improved. Last but not least, Server Push enables the proactive addition of response into the browser cache which make feel that HTTP 2 better than HTTP 1.1 among developers and end users.

1. Objects & its internal representation in JavaScript:

Object is non-primitive or reference data type (i.e. dynamic in nature, do not have a fixed size) and also it is a data structure. Primitive data type (Number, String, Boolean, Null, Undefined and Symbol) are stored by single value each depending on their types.

Object Literal – it is simple way to write a object. It is having the “Key: Value” pair. The followings are explained it easily. Here, Key is also considered as variable Name.

Example,

let FSD = {

StudentName : “Vinoth”,

StudentAge : 35,

StudentId : “FSD-143”

Hobbies : [‘singing’, ‘reading’, ‘surfing out’]

}

Or

Obj.property

Obj[“property”]

Curly braces {} to define a object in Javascript is called Object Literal. “Key”: value pair data type. Can store primitive and other object, array anything we can store, except, Key is case-sensitive. The keys can be variable or function and are called properties and methods, respectively.

We can access the Object properties through DOT NOTATION (.) and BRACKET NOTATION ()

In the list of data, every line of data ended with comma (,) except ending line.

Example,

Console.log (FSD.StudentName); // output Vinoth

Console.log (FSD [‘StudentAge’]); // output 35

Instead of Dot Notation, Bracket Notation can be accessible in all kind of Object properties (key).

Example,

“#Gender#” : Male // within Quotes we can give any variable name (key)

Console.log (FSD.#Gender#) //output undefined or JavaScript keyword error occurred.

Console.log(FSD[‘#Gender#’]) // output Male

Key Name Identifier can be contained of alphabetical letters, Numbers, $, underscore (\_).

StudentName : ‘vinoth’ // correct

007StudentName : ‘vinoth’ //key identifier not allowed

Student\_Name : ‘Vinoth’ //correct

$Student$Name : ‘Vinoth’ //correct

And Key name with JavaScript keyword and Space <space> are not allowed.

Student Name : ‘Vinoth’ // space error

Object Prototype – Each Object in JavaScript has a built-in property.