**HTTP/2 AND HTTP/1.1:**

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.

**Difference between HTTP/1.1 and HTTP/2:**

|  |  |
| --- | --- |
| **HTTP/1.1** | **HTTP/2** |
| The standardized protocol | A protocol for greater performance |
| Its works on the textual format. | It works on the binary protocol. |
| It uses requests resource Inlining for use getting multiple pages. | It uses PUSH frame by server that collects all multiple pages. |
| It supports connection reuse i.e. for every TCP connection there could be multiple requests and responses, and pipelining where the client can request several resources from the server at once. However, pipelining was hard to implement due to issues such as head-of-line blocking and was not a feasible solution. HTTP2 Protocol | Uses multiplexing, where over a single TCP connection resources to be delivered are interleaved and arrive at the client almost at the same time. It is done using streams which can be prioritized, can have dependencies and individual flow control. It also provides a feature called server push that allows the server to send data that the client will need but has not yet requested. HTTP3 Protocol |

HTTP2 Vs. HTTP1 is not a debate at all. HTTP2 is much faster and more reliable than HTTP1. HTTP1 loads a single request for every TCP connection, while HTTP2 avoids network delay by using multiplexing.

**BLOG ABOUT OBJECTS AND ITS INTERNAL REPRESENTATION IN JAVASCRIPT;**  
  
 Objects, in JavaScript, is its most important data-type and forms the building blocks for modern JavaScript. These objects are quite different from JavaScript’s primitive data-types **(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.  
  
let school = {

name : “Vivekananda School”,

location : “Delhi”,

established : “1971”

}

In the above example “name”, “location”, “established” are all “keys” and “Vivekananda School”, “Delhi” and 1971 are values of these keys respectively.

Each of these keys is referred to as properties of the object. An object in JavaScript may also have a function as a member, in which case it will be known as a method of that object.