|  |  |
| --- | --- |
| **HTTP 1.1** | **HTTP 2** |
| It works on the textual format | It works on the binary protocol. |
| There is head of line blocking that blocks all the requests behind it until it doesn’t get its all resources. | It allows multiplexing so one TCP connection is required for multiple requests. |
| 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 loads resources by splitting data into binary-code messages and numbering these messages so that the client knows which stream each binary message belongs to. |
| It compresses data by itself. | It uses HPACK for data compression. |
|  | HTTP2 allows servers to “push” responses proactively into client caches. |

**Difference between HTTP1.1 vs HTTP2**

**Objects and its internal representation in Javascript**

Objects forms the building blocks for modern JavaScript. 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.

Every object has some property associated with some value. These values can be accessed using these properties associated with them.

Example : var student = { “name” : “abc” , “age” : 18 ,”marks” : 80 }

Here student is object , “name” is key and “abc” is its value.

The value can be retrieved in two ways as following :

student.age (or) student[“age”]

Which give output value as 18.