| **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. |  |  |
| * It uses requests resource Inlining for use getting multiple pages | * It uses PUSH frame by server that collects all multiple pages |  |  |
| * It compresses data by itself. | * It uses HPACK for data compression. |  |  |
| * There is no flow priority. | * Improved flow priority mechanism are used. |  |  |
| * No header compression | * Metadata compression using improved algorithms. |  |  |

**Day Task-1**

1.Difference between HTTP1.1 vs HTTP2

2. Objects and its internal representation in JavaScript

* An object is a collection of properties, and a property is an association between a name (or key) and a value. A property's value can be a function, in which case the property is known as a method.
* Objects in JavaScript, just as in many other programming languages, can be compared to objects in real life.

Internal representation:

* Objects are important data types in JavaScript. Objects are different than primitive datatypes (i.e. number, string, Boolean, etc.). Primitive data types contain one value but Objects can hold many values in form of Key.
* 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.

For example: Person, car, pen, bike, Personal Computer, Washing Machine etc.

Take the case of cars.

All cars have the same properties, but the property values differ from car to car. All cars have the same methods, but the methods are performed at different times.



Let’s have an example of my favourite merc car and list out its properties (Features):

1. Make: Mercedes
2. Model: C-Class
3. Color: White
4. Fuel: Diesel
5. Weight: 850kg
6. Mileage: 8Kmpl
7. Rating: 4.5

Taking the above as reference, I'll stress up on objects, Object properties and Methods.

**Objects:**

The following code assigns a simple value (Mercedes) to a variable named car:

var car = "Mercedes";

Objects are variables too. But objects can contain many values.

The following code assigns many values (Mercedes, C-class, White and so on) to a variable named Car:

var car = {Make: “Mercedes”, Model: “C-Class”, Color: “White”, Fuel: Diesel, Weight: “850kg”, Mileage: “8Kmpl”, Rating: 4.5};

**Object Properties**

The name: values pairs (in JavaScript objects) are called **properties**.

var car = {Make: “Mercedes”, Model: “C-Class”, Color: “White”, Fuel: Diesel, Weight: “850kg”, Mileage: “8Kmpl”, Rating: 4.5};

From the above snippet, let’s have a look what falls under property and property value:



The object properties can be different primitive values, other objects and functions.

Properties can usually be changed, added, and deleted, but some are read only.

**Object methods:**

An object method is an object property containing a function definition.

i.e.,

Let’s assume to start the car there will be a mechanical functionality.

function () {return ignition. On}

and so similar is to stop/brake/headlights on & off, etc.