|  |  |
| --- | --- |
| HTTP 1.1: | HTTP 2: |
| HTTP 1.1 originates from HTTP (Hypertext transfer protocol) that runs on top of the internet TCP/ I.P (Transmission control protocol). It will produce faster delivery of web pages than the original http and reduces web traffic.  HTTP/1.1 loads resource one after the other, so if one resource cannot be loaded, it blocks all the other resources behind it | The primary goals for HTTP/2 are to reduce latency by enabling full request and response multiplexing.  HTTP/2 does not modify the application semantics of HTTP in any way. All the core concepts found in HTTP 1.1, such as HTTP methods, status codes, URIs, and header fields, remain in place. Instead, HTTP/2 modifies how the data is formatted (framed) and transported between the client and server, both of which manage the entire process, and hide application complexity within the new framing layer. As a result, all existing applications can be delivered without modification.  In contrast, HTTP/2 is able to use a single TCP connection to send multiple streams of data at once so that no one resource blocks any other resource. |

Objects in JavaScript:

The Object type represents one of JavaScript's data types. It is used to store various keyed collections and more complex entities. Objects can be created using the Object() constructor or the object initializer / literal syntax.

7 primitive data types:

* Number: Floating point number used for decimals and integers.

Example: Let age = 23;

* String: sequence of character used for text

Example: Let final name = “Jonas”;

* Boolean is a logical type that can be true or false and used for making decisions.

Example: Let fullage = true;

* Undefined: a value taken by a variable that is not yet defined empty value
* Null: It also means an empty value.
* Symbol (ES2015): Value that is unique and cannot be changed.
* Big int (ES2020): Larger integers than the number type can hold.

BLOG ABOUT OBJECTS AND THEIR INTERNAL REPRESENTATION OF JAVASCRIPT:

In simple terms. “A JavaScript object is a collection of named values having state and behaviour (properties and method)”.

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 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 a reference, I'll stress objects, Object properties and Methods.

Graphical user interface, text, application

Description automatically generated

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

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

**Object Properties:**

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

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

|  |  |
| --- | --- |
| PROPERTY | PROPERTY VALUE |
| Make | Mercedes |
| Model | C-Class |
| Colour | White |
| Fuel | Diesel |
| Weight | 850kg |
| Mileage | 8kmpl |
| Rating | 4.5 |

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 mechanical functionality.

Eg. Of code: **function(){return ignition.on}**

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

**Conclusion:**

In a nutshell, “methods are actions that can be performed on objects”