1. blog about objects and its internal representation in Jav
2. Write a blog on Difference between HTTP1.1 vs HTTP2

HTTP.1

* HTTP/2 is much faster and more efficient than HTTP/1.1. One of the ways in which HTTP/2 is faster is in how it prioritizes content during the loading process.
* HTTP1.1 is works on the textual format, whereas HTTP/2 is works on the binary protocol.
* HTTP/1.1 loads resources one after the other, so if one resource cannot be loaded, it blocks all the other resources behind it. 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. HTTP/2 does this by splitting data into binary-code messages and numbering these messages so that the client knows which stream each binary message belongs to.
* Small files load more quickly than large ones. To speed up web performance, both HTTP/1.1 and HTTP/2 compress HTTP messages to make them smaller. HTTP/1.1 compresses data by itself, whereas uses HPACK for data compression. This eliminates a few bytes from every HTTP packet.
* HTTP/1.1 requests resource Inlining for use getting multiple pages. But HTTP/2 uses PUSH frame by server that collects all multiple pages

1. Write a blog about objects and its internal representation 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: value pair. 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.

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

Make: BMW

Model: X5

Color: Black

Fuel: Diesel

Weight: 850kg

Mileage: 8Kmpl

Rating: 4.5

1. codekata practice

<https://www.guvi.in/code-kata-main?concept=absolute%20beginner>

let array = userInput[0].split('');

let sum = 0;

for(let i=0; i<array.length; i++){

sum = sum + parseInt(array[i]);

}

console.log(sum);

});