1. Write a blog on Difference between HTTP1.1 vs HTTP2

|  |  |
| --- | --- |
| 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 in lining 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. |
| It was created in 1997 | It was created in 2015 |

2. Write a blog about objects and internal representation in JavaScript

Objects are important data types in JavaScript. Objects are non primitive data types. That allows you to store multiple collections of data. It is used to store various keyed collections and more complex entities. Objects can be created using the Object().

Example for JavaScript Object:

const student = {

firstName: 'ram',

class: 10

};

A JavaScript object has properties associated with it. A property of an object can be explained as a variable that is attached to the object. Object properties are basically the same as ordinary JavaScript variables, except for the attachment to objects. The properties of an object define the characteristics of the object. You access the properties of an object with a simple dot-notation:

objectName.propertyName

Properties of JavaScript objects can also be accessed or set using a bracket notation . Objects are sometimes called associative arrays, since each property is associated with a string value that can be used to access it.

The syntax for adding a property to an object is :

ObjectName.ObjectProperty = propertyValue;

The syntax for deleting a property from an object is:

delete ObjectName.ObjectProperty;

The syntax to access a property from an object is:

objectName.property // Car.Make

objectName["property”] // Car["Make"]

objectName [expression] // x = "Make"; Car[x]

Java Script properties are “Properties are the values associated with a JavaScript object”.

Create JavaScript Object with Constructor

Constructor is nothing but a function and with help of new keyword, constructor function allows to create multiple objects of same as shown below

function Vehicle(name, maker) {

this.name = name;

this.maker = maker;

}

let car1 = new Vehicle(’Fiesta’, 'Ford’);

let car2 = new Vehicle(’Santa Fe’, 'Hyundai’)

console.log(car1.name); //Output: Fiesta

console.log(car2.name); //Output: Santa Fe

Using the JavaScript Keyword new

The following example also creates a new JavaScript object with four properties:

Example

var person = new Object();  
person.firstName = “John”;  
person.lastName = “Doe”;  
person.age = 50;  
person.eyeColor = “blue”

Using the Object.create method

Objects can also be created using the Object.create() method. This method can be very useful, because it allows you to choose the prototype object for the object you want to create, without having to define a constructor function.