|  |  |
| --- | --- |
| **HTTPS1.1** | **HTTPS2** |
| * It compresses data by itself      * HTTP/1.1 loads resources one after the other, so if one resource cannot be loaded, it **blocks** all the other resources behind it * It works on **textual** formal * It uses requests resource **Inlining** for getting multiple pages. * Rendering time of http1.1 is **more** compared to https2 | * It compresses data using **HPACK** * HTTP/2 is able to use a single TCP connection to send multiple data at once (**Multiplexing**), so that it doesn’t blocks any other resources. * It works on **binary** format * It uses **PUSH** frame by server that collects all multiple pages * Rendering time for http2 is **less** compared to https1.1 |

**DAY 1 – TASK**

TASK 1- Difference between HTTPS1.1&HTTPS2.

TASK 2- Objects and its Internal Representation

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

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

Take the case of cars.

All cars have the same properties, but the property values differ from car to car.

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

1. Make: Skoda
2. Model: Slavia
3. Colour: White
4. Fuel: Diesel
5. Weight: 850kg
6. Mileage: 18Kmpl

Taking the above as reference, I'll apply it on objects and object properties,

**1)Objects:**

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

var car = "Skoda";

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

The following code assigns **many values** (Skoda, Slavia, White and so on) to a **variable** named Car:

var car = {

Make: “Skoda”,

Model: “Slavia”,

Colour: “White”

Fuel: “Diesel”

Weight: “850kg”

Mileage: “18Kmpl

}

The values are written as **name: value** pairs (name and value separated by a colon).

Syntax:

var <object-name> = {key1: value1, key2: value2.....keyN: valueN};

So, conclusion and definition for JS objects is “JavaScript objects are containers for named values”.

## ****Object Properties****

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

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.

**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

//or

objectName["property”]

So, Conclusion and simple definition for Java Script properties is “Properties are the values associated with a JavaScript object