|  |  |  |
| --- | --- | --- |
| **Feature** | **HTTP/1.1** | **HTTP/2** |
| **Connection Handling** | Separate connections for each resource. | Multiplexes multiple resources over a single connection. |
| **Header Compression** | Headers sent in plaintext, leading to larger overhead. | Utilizes header compression, reducing header size and improving efficiency. |
| **Server Push** | Not supported. Clients request resources individually. | Allows servers to push resources proactively to the client. |
| **Prioritization** | No built-in prioritization. All requests are treated equally. | Supports prioritization, fetching more critical resources first. |
| **Flow Control** | Lacks built-in flow control. | Implements advanced flow control, preventing congestion. |
| **Multiplexing** | Limited multiplexing capabilities, leading to slower page loading. | Efficiently multiplexes multiple streams over a sin |

**1) Write a blog on the Difference between HTTP1.1 vs HTTP2**

### ****2)Write a blog about objects and its internal representation in Javascript.****

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 the 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.

**JavaScript Object:**

JavaScript object is a collection of key-value pairs, where each key is a string (or a symbol) and each value can be of any data type, including other objects.

var myCar = new Object();

myCar.make = 'Suzuki';

myCar.model = 'Altros';

myCar.year = 1978;

myCar.wheels = 2;

After creating myCar object, the value inside the object can be accessed using keys.

i.e.

myCar.year

Output: 1978

These values can be accessed using brackets notation also.

myCar[year]

Output: 1978

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;

## Using the JavaScript Keyword new

var person = new Object();

person.firstName = “John”;

person.lastName = “Doe”;

person.age = 50;

person.eyeColor = “blue”;Create JavaScript Object with Constructor

## Create JavaScript Object with Constructor

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