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

Ans:

Both HTTP/1.1 and HTTP/2 are two different versions of HTTP protocol used for the data communication between client and server, in short it is a client-server protocol. Let’s see some of the key differences among these versions.

| HTTP/1.1 | HTTP/2 |
| --- | --- |
| It does not support multiplexing, that is multiple requests cannot be sent simultaneously through a single TCP connection. We want to wait for the previous requests to complete for making new requests to the server. So, there is latency. | It allows multiplexing, that is multiple requests can be sent to the server and the server responds to the client with multiple streams of data over a single TCP connection. As the data is sent as multiple streams, the latency is reduced so well. |
| Headers are not compressed, it leads to increased overhead because each request and response include full header information. | As the headers are compressed, the amount of data transfer is limited and that increases the efficiency. |
| As Text-based format is used for messages, it is efficient for machine processing. | It uses the binary format for data transformation. Hence, it is easier for machine parsing. |
| It has no prioritization for requests. As all requests happen synchronously, so that we cannot make any prioritization on it and that causes the head-of-line blocking. | As there is prioritization, the import requests are handled for delivering specific resources that reduce the page load latency. |
| There is no server push. Hence, each time we need to make explicit requests to the server for the resources. | It supports the server push that enables the server to push the resources to the client without explicit requests from the client that reduces the multiple round-trip requests. |
| To handle multiple resources at a time, we need to create multiple connections which makes the resource intensive. | It handles a single, multiplexed connection that reduces the multiple connection and improves the resource utilization. |
| It is fully backward-compatible with previous versions of HTTP protocol | As it is built with backward compatibility, in some situations it may not work optimally over networks that are not optimized for its feature. |

1. Write a blog about objects and its internal representation in Javascript.

Ans:

For a sense we take a real world object like a dice as an example for it. In this case, the property of the cube is its six faces and each face is numbered. The action associated with it is when it is rolled, its one face reveals the specific number for the game move. So in reality, objects have properties and corresponding actions.

In the programming world this concept is incorporated for simulating real world objects. By definition, an object is a collection of related data and methods. In javascript, the objects are represented(syntax) as key-value pairs inside the curly braces. The values to the keys may be any data type and methods. These objects can contain one or more key-value pairs.

The person object can be created as follows:

var person = {

name: "John Doe",

age: 35,

profession: "software engineer",

introduce: function (){

return `I am ${this.name} working as a ${this.profession}.`

}

}

console.log(person.introduce())