### 1. Difference between http1.1 and http2

|  |  |
| --- | --- |
| **http1.1** | **http2** |
| 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. |

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

JavaScript is designed on a simple object-based paradigm. An object is a collection of properties, and a property is an association between a name (or *key*) and a value. A property’s value can be a function, in which case the property is known as a method.

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. We can access the properties of an object with a simple dot-notation:

* 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 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.

The syntax for adding a property to an object is :

**ObjectName.ObjectProperty = propertyValue;**

## ****JavaScript’s internal representation of Objects:****

A simple diagram is probably the best way to give a quick overview of the object representation in Javascript.

