**Objects And Its Representation:**

**Objects:**

Objects in JavaScript may be defined as an unordered collection of related data, of primitive or reference types, in the form of “key: value” pairs. These keys can be variables or functions and are called properties and methods, respectively, in the context of an object.

Every object has a parent object called ***Prototype.***

**Internal Representation:**

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.

Javascript objects can be accessed using 2 methods.

**1. Dot Method**: Used for normal accessing and printing the object properties.

* Syntax: ObjectName. keyName
* Example: console.log(Student.Name);

**2. Box/Bracket Method:** Used for printing the Object properties one by one. Here the keys are enclosed in double quotes(“”).

* Syntax: ObjectName[“keyName”]
* Example: console.log(Student[“Name”]);

**HTTP 1.1 VS HTTP 2**

**What is HTTP?**

HTTP stands for Hyper Text Transfer Protocol. It is the basis of all the web applications. HTTP is a layer control protocol for transmitting hypermedia documents such as graphics, audio, video, plain text and hyperlinks on the World Wide Web. HTTP is the method computers and servers use to request and send information. HTTP is a ***stateless protocol*** that means the target server does not support information between two requests.

**Difference between HTTTP/1.1 & HTTP/2**

***Multiplexing:***

* **HTTP/1.1** loads resources one after the other, so if one resource cannot be loaded, it blocks all the other resources behind it.
* **HTTP/2** is able to use a single TCP connection to send multiple streams of data at once so that no one resource blocks any other resource. HTTP/2 does this by splitting data into binary-code messages and numbering these messages so that the client knows which stream each binary message belongs to.

***Server push****:*

* A server only serves content to a client device if the client asks for it. However, this approach is not always practical for modern web-pages.
* HTTP/2 solves this problem by allowing a server to "push" content to a client before the client asks for it.

***Header compression****:*

* Small files load more quickly than large ones. To speed up web performance, both HTTP/1.1 and HTTP/2 compress HTTP messages to make them smaller.
* HTTP/2 uses a more advanced compression method called HPACK that eliminates redundant information in HTTP header packets. This eliminates a few bytes from every HTTP packet.

***Prioritization:***

* Prioritization refers to the order in which pieces of content are loaded. Prioritization affects a webpage's load time.
* Stream Prioritization is a technique that allows customization of requests relative weight to optimize application performance and solves potential requests competition for the same resource