**Difference between HTTP1.1 vs HTTP2**

* The basic difference is HTTP2 is much faster and more reliable than HTTP1.1
* HTTP1.1 loads a single request for every TCP connection, while HTTP2 avoids network delay by using multiplexing.
* HTTP1.1 supports connection reuse i.e. for every TCP connection there could be multiple requests and responses . Whereas HTTP2 Uses multiplexing, where over a single TCP connection resources to be delivered are interleaved and arrive at the client almost at the same time.
* HTTP 1.1 can define 24 status codes, error reporting is quicker and more efficient. It also remains the same for HTTP2
* HTTP1.1 is relatively secure since it uses digest authentication, NTLM authentication. Whereas HTTP2 is better equipped to deal with the security issues due to new TLS features like connection error of type Inadequate Security.
* HTTP1.1 Expands on the caching support by using additional headers like cache-control, conditional headers like If-Match and by using entity tags, whereas, With the server push feature if the client finds the resources are already present in the cache, HTTP2 can cancel the pushed stream.
* HTTP1.1 provides faster delivery of web pages and reduces web traffic. But HTTP2 utilizes multiplexing and server push to effectively reduce the page load time by a greater margin along with being less sensitive to network delays.

**Objects and its internal representation in JavaScript**

* Objects, in JavaScript, is it’s most important data-type and forms the building blocks for modern JavaScript.
* Objects are more complex and each object may contain any combination of these primitive data-types as well as reference data-types.
* 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
* the properties of an object can be accessed with a simple dot-notation
* 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.
* Objects are mutable: They are addressed by reference, not by value.
* A property is a “key: value” pair, where a key is a string (also called a “property name”), and value can be anything.
* Example; let car = {  
  name : “BMW”,  
  model: “720 D”,  
  fuel: “Diesel”  
  }