**DAY 1: ASSESSMENT**

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

Two variations of the Hypertext Transfer Protocol, which runs the internet, are HTTP/1.1 and HTTP/2. Understanding their key distinctions might help you comprehend how the web has developed for greater performance and speed.

**1] Multiplexing:**

**HTTP1.1:** Imagine a waiter at a restaurant taking orders from a group of diners one at a time. Similar to a waiter receiving orders one at a time, web browsers and servers converse with one another in HTTP/1.1. If there are numerous items to order (such as photos, scripts, and stylesheets), this may take a while.

**HTTP/2:** With HTTP/2, serving customers is more effective since the waiter may carry several orders on a tray at once. Thus, numerous elements of a web page can be requested and loaded by web browsers at once, speeding up website loading.

2**] Header compression:**

Think of headers in **HTTP/1.1** as labels on packages that describe the contents. These labels are large and unnecessary in HTTP/1.1, similar to printing the same delivery address on every package. Time and resources are wasted on this.

**HTTP/2:** employing HTTP/2 is similar to employing smart labels that can recognise what is in a box. These labels are compressed, resulting in less data being transferred back and forth and accelerating communication.

3] **Server Push:**

**HTTP/1.1:** Picture yourself in a restaurant having to order each dish separately, even if it is a standard side dish that is included with your order. It is ineffective.

**HTTP/2:** HTTP/2 is similar to a restaurant where the chef orders your food for you without your request. It is time-saving and proactive.

4] **Connection management:**

When you interact with the waiter over **HTTP/1.1**, it's similar to opening and terminating a new connection. This requires time and work.

**HTTP/2:** There is reduced overhead and more rapid processing since HTTP/2 maintains the connection open, much like having an ongoing discussion with the waiter.

5] **Backward Compatibility:**

**HTTP/1.1:** It's like using an antiquated, universally understood language, despite not being the most effective.

**HTTP/2:** You can understand and communicate with people who speak HTTP/1.1 while speaking HTTP/2, a new language. It's like speaking two languages online.

1. **Objects and its internal representation in Javascript:**

Objects are a fundamental data type in JavaScript that are used to store and modify data. They are a group of key-value pairs, where the keys (also known as properties) are strings (or symbols) and the values can be other objects or any other data type. You can develop more effective code and choose better design options if you know how JavaScript objects are internally represented.

* **Properties and Values:** Objects consist of properties and their corresponding values. Properties can be thought of as keys that uniquely identify a value within an object.
* **Object Prototype:** Every object in JavaScript is associated with a prototype object, which provides a set of properties and methods that the object can inherit. This concept is central to JavaScript's prototypal inheritance.
* **Property Descriptors:** Each property in an object has associated property descriptors that define its behaviour. These descriptors include information such as whether the property is writable, enumerable, and configurable.
* **Hidden Classes and Inline Caching (V8 Engine):** JavaScript engines like V8 (used in Chrome) use hidden classes to optimize property access. When you create an object and add properties to it, V8 dynamically creates hidden classes to represent these objects' shapes. This allows for faster property access through inline caching, as objects with the same shape share the same hidden class.
* **Dictionaries (Spider Monkey Engine):** Other JavaScript engines like Spider Monkey (used in Firefox) use a different approach, where they store properties in a dictionary-like structure. This allows for more flexibility but may be less optimized for property access than V8's hidden classes.