**1. Reading Exercises.**

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

The HTTP1.1 was introduced in the year 1997 just after its previous version came into existence at 1996, unlike its precursors. The reason being, the internet ecosystem was going through a rapid change and this was initial phase of the internet revolution. Over almost two decades later HTTP2.0 was introduced with features which simplified the issues with HTTP1.1.

One of the major feature that differentiates HTTP2.0 from HTTP1.1 is the binary framing layer. Unlike HTTP1.1, HTTP2.0 uses a binary framing layer instead of text. As we know binary codes are more straightforward which saved a lot of processing time. While making sure that its HTTP semantics remain untamed.

Secondly, HTTP2.0 is fully multiplexed. To illustrate, HTTP1.1 loads resources one after the other, so if one resource cannot be loaded, it blocks all the other resources behind it. In contrast, HTTP2.0 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. HTTP2.0 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. Although the number of TCP connections in HTTP1.1 were increased to a maximum of 6 at later stages, it couldn’t function as efficient as HTTP2.0 as it does now.

HPACK feature in HTTP2.0 separated header data from the request data enabling us to reuse the header data which was repeated under every request in HTTP1.1. These above are some of the many features that reduced the processing time in-turn giving us the fast internet connection that we know of today.

**3. Write a blog about objects and its internal representation in JavaScript.**

Objects and arrays are considered to be special in JavaScript. Particularly objects are JavaScript’s most important data-type and forms the building blocks for modern JavaScript.

An object, is a reference data type. Variables that are assigned a reference value are given a reference or a pointer to that value. That reference points to the location in memory where the object is stored. While JavaScript’s primitive data-types stores the value acting as container.

Loosely speaking, 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. The functions that are respective to these objects are called the methods.

**Syntax –**

var object\_name = {key:”value”, key:”value”, key:”value”};

For Eg. If your object is a student, it will have properties like name, age, address, id, etc and methods like updateAddress, updateNam, etc.

**Accessing a property in an object-**

object\_name.keyOfTheValueWeWantToAccess;

**For Example** it can be something like user.purchases;

**4. Codekata Practice.**