**Day -1**

**Introduction to Browser & web**

**Http:**

* HTTP stands for **Hypertext Transfer Protocol**.
* It is use to exchange any data on the Web.
* Allows web browsers and servers to communicate and exchange information.

request

Server

Network

Client

response

(Browser) (Internet) (web server)

**Example:**

Allow to exchange of text, images, videos, and other content when you browse websites.

server

Client

**Activity**

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

|  |  |  |
| --- | --- | --- |
| **Sl.No** | **Http1.1** | **Http2** |
| **1** | Supports only one request at a time | allowing multiple simultaneous requests and responses on a single connection |
| **2** | Headers are not compressed, resulting in increased data overhead | Implements header compression, reducing the amount of data transmitted |
| **3** | Uses a text-based protocol, which can be less efficient in terms of parsing and processing. | Adopts a binary protocol, offering a more compact representation of data and faster parsing. |
| **4** | Processes requests in the order they are received, without built-in prioritization. | Introduces stream prioritization, enabling clients to assign priority levels to different resources for optimized loading. |
| **5** | Lacks native support for server push. | Supports server push, allowing servers to push resources to clients before they are explicitly requested. |
| **6** | Requires multiple connections for parallelism, potentially leading to resource contention. | Achieves parallelism over a single connection, reducing the need for multiple connections and avoiding contention issues. |
| **7** | Multiple round trips are often needed for request/response exchanges. | Minimizes round trips through features like header compression and multiplexing, enhancing efficiency. |
| **8** | Fully backward-compatible with HTTP/1.0. | Designed to be backward-compatible with HTTP/1.1, allowing gradual adoption |
| **9** | May suffer from head-of-line blocking, where slow-loading resources delay others | Mitigates head-of-line blocking through multiplexing and prioritization, improving overall resource loading. |
| **10** | Ubiquitous and widely adopted but has inherent limitations. | Adoption requires server and client support, but its performance benefits incentivize migration. |

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

**Answer :**

In JavaScript, almost "everything" is an object. So object are **king of java script.**

* Arrays are always objects
* Functions are always objects
* Booleans can be objects
* Numbers can be objects
* Strings can be objects
* Objects are always objects

objects are a fundamental data type that allows you to store and organize data. Objects in JavaScript are dynamic, meaning you can add or remove properties and methods to/from them at any time.

The internal representation of objects in JavaScript involves a few key concepts:

1. **Prototypes:**
   * Objects are linked to other objects known as prototypes. If a property isn't found in an object, JavaScript looks up the prototype chain until it finds the property or reaches the end.
2. **Properties and Descriptors:**
   * Properties of objects have descriptors that define attributes like whether a property is writable, enumerable, or configurable.
3. **Dynamic Nature:**
   * Objects are dynamic, allowing properties and methods to be added or removed during runtime.
4. **Constructor Functions:**
   * Constructor functions help create objects with a specific structure when used with the **new** keyword.
5. **Object Models:**
   * JavaScript supports different object models, including the prototype-based model and the class-based model introduced in ES6.