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

Answer:

|  |  |
| --- | --- |
| HTTP 1 | HTTP 2 |
| HTTP/1uses a single connection to handle one request at a time, leading to head-of-line blocking issues. | HTTP/2 supports multiplexing, allowing multiple requests and responses to be sent in parallel over a single connection. |
| HTTP/1 headers are sent in plain text, which increases overhead, especially for repetitive headers. | . HTTP/2 compresses headers, reducing the amount of data transmitted and improving efficiency |
| HTTP/1uses a text-based protocol, which is human-readable but less efficient in terms of parsing and processing. | HTTP/2 uses a binary framing layer that is more compact and faster to parse. |
| HTTP/1lacks this feature, leading to potential delays for essential content. | HTTP/2 allows for stream prioritization by assigning weights to different streams, ensuring critical resources are delivered first. |
| In HTTP/1, each request/response contains duplicated headers, which contributes to overhead.. | HTTP/2 enables header field compression and multiplexing, reducing redundancy and improving data transfer efficiency |
| HTTP/1 does not have built-in encryption, leading to potential security vulnerabilities when transmitting sensitive data. However, it can be used with HTTPS (HTTP Secure) to provide encryption through SSL/TLS protocols. | HTTP/2 is optimized for mobile devices, offering reduced latency and improved performance by employing features such as multiplexing and header compression. This results in faster page loads and a better user experience on mobile networks. |
| HTTP/1 uses a text-based protocol where messages between the client and server are in plain text, making it easy for developers to read and understand the communication but potentially less efficient compared to binary protocols. | HTTP/2 includes flow control mechanisms that manage the data transfer between the server and client. This prevents overwhelming the recipient with data it cannot process quickly, improving performance and stability. |
| HTTP/1 uses URIs to identify and locate resources on the web. The URI, commonly referred to as a URL, specifies the address of a resource to be retrieved by the client. | HTTP/2 can manage dependencies between different parts of a webpage, allowing for better resource loading prioritization. This helps in loading crucial resources faster and more efficiently. |
| HTTP/1 defines a range of status codes indicating the outcome of a client's request, such as 200 OK (successful), 404 Not Found (resource not found), 500 Internal Server Error (server error), among others. | HTTP/2 introduces server push, allowing the server to send multiple responses to a single client request. This feature enables the server to anticipate and push resources the client hasn't yet requested, optimizing performance. |

2. Write a blog about 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. These objects are quite different from JavaScript’s primitive data-types(Number, String, Boolean, null, undefined and symbol) in the sense that while these primitive data-types all store a single value each (depending on their types)Objects are more complex and each object may contain any combination of these primitive data-types as well as reference data-types.  
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 or pointer points to the location in memory where the object is stored. The variables don’t actually store the value.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.

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