DAY /01

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

ANS:

HTTP (Hypertext Transfer Protocol) is the foundation of data communication on the internet. It is the protocol used by web browsers and servers to communicate and exchange information. HTTP has gone through several revisions, with HTTP/1.1 being the most widely used version until recently. In 2015, the latest version, HTTP/2, was released, promising to improve the web browsing experience.

|  |  |
| --- | --- |
| HTTP 1 | HTTP 2 |
| **1.Connection Management**   * **In HTTP/1, each request and response is sent over a separate connection, which can result in increased latency and slower page loading times.** | **1. Connection Management**   * **In contrast, HTTP/2 uses a single connection for multiple requests, reducing the number of round trips required to load a page and improving performance.** |
| 1. **Binary Protocol**  * **while HTTP/1 is a text-based protocol.** | **2.Binary Protocol**   * **HTTP/2 is a binary protocol,** * **HTTP/2 uses a more efficient encoding method, which reduces the amount of data that needs to be sent between the client and server. This results in faster page loading times and a better overall browsing experience.** |
| 1. **Server Push**  * **In HTTP/1.1 server push is not natively supported. Instead, a web page must request each resource (such as images, CSS files, and JavaScript files) individually. This means that a web page may require multiple round trips between the client and server to load all the necessary resources. This can result in slower page load times and increased latency.** | 1. **Server Push**  * **HTTP/2 introduces a new feature called server push, which allows the server to send multiple responses for a single request. This means that the server can send the main HTML file, along with additional resources such as CSS and JavaScript files, without waiting for the client to request them. This can result in faster page loading times and a better browsing experience, particularly for large websites with many resources.** |

|  |  |
| --- | --- |
| 1. **Header Compression**  * **In HTTP/1.1, headers can account for a significant portion of the total data sent between the client and server, which can impact page loading times. With header compression,** | 1. **Header Compression**  * **HTTP/2 also includes header compression, which reduces the size of the headers sent between the client and server.** * **HTTP/2 can reduce the size of headers by up to 80%, resulting in faster page loading times.** |
| 1. **Security**  * **HTTP/1.1 does not require encryption, and many websites still use plain HTTP.** | 1. **Security**  * **TTP/2 requires the use of SSL/TLS encryption, which provides an additional layer of security for web browsing.** * **With HTTP/2, all communication is encrypted, which protects user data and helps prevent unauthorized access** |

Conclusion:

HTTP/2 is a significant improvement over HTTP/1.1, offering faster page loading times, improved performance, and enhanced security. While HTTP/1.1 remains in use today, it is clear that HTTP/2 is the future of web browsing. As more websites adopt HTTP/2, users can expect a faster, more secure, and better overall browsing experience.

Top of Form

2.Write a blog about objects and its internal representation in Javascript?

ANS: In JavaScript, an object is a collection of properties that has a set of key-value pairs. Properties can be added, removed, and modified at runtime. An object can also have methods, which are functions that are associated with the object and can be called on the object.

JavaScript objects are represented internally as key-value maps. When we create an object, the JavaScript engine allocates memory to store the object and its properties. Each property is stored as a key-value pair in the object's map. The key is the property name, and the value is the value of the property.

Object Methods

In addition to properties, objects in JavaScript can also have methods. Methods are functions that are associated with an object and can be called on the object.

When we call a method on an object, the method is executed in the context of the object. This means that the method can access the object's properties and modify them if necessary.

Objects are a fundamental part of JavaScript and are used extensively in modern web development. Understanding their internal representation is essential for working with them effectively. In this blog, we have explored how JavaScript objects are represented internally as key-value maps and how we can use them to store and manipulate collections of related data.

Top of Form

Top of Form

Bottom of Form