1.difference between HTTP1. 1 vs HTTP2

HTTP1 - Developed by Timothy Berners-Lee in 1989 as a communication standard for the World Wide Web.

HTTP is a top-level application protocol that exchanges information between a client computer and a local or remote web server.

is that can only load requests one at a time, one request per one TCP connection.

In this process, a client sends a text-based request to a server by calling a method like `GET` or `POST`. In response, the server sends a resource like an HTML page back to the client.

HTTP2 - HTTP2 is the binary framing layer,

which can be thought of as a part of the application layer in the internet protocol stack. As opposed to HTTP1, which keeps all requests and responses in plain text format, HTTP/2 uses the binary framing layer to encapsulate all messages in binary format.

while still maintaining HTTP semantics, such as verbs, methods, and headers. An application level API would still create messages in the conventional HTTP formats, but the underlying layer would then convert these messages into binary. This ensures that web applications created before HTTP2 can continue functioning as normal when interacting with the new protocol.

- HTTP2 is binary, instead of textual
- HTTP2 is fully multiplexed, instead of ordered and blocking
- HTTP2 can, therefore, use one connection for parallelism
- HTTP2 uses header compression to reduce overhead

2 .http version history

The first documented version of HTTP was **HTTPV0.9**(1991).

Dave Raggett led the HTTP Working Group (HTTP WG) in 1995 and wanted to expand the protocol with extended operations, extended negotiation, richer meta-information, tied with a security protocol which became more efficient by adding additional methods and header fields. RFC 1945 officially introduced and recognized **HTTP V1.0 i**n 1996.

That same web hosting company reported that by June 1996, 65% of all browsers accessing their servers were **HTTPv1.1** compliant. The **HTTP** v**1.1** standard as defined in RFC 2068 was officially released in January 1997.

HTTPv2 is a more efficient expression of HTTP's semantics "on the wire", and was published in 2015, and is used by over 50% of websites; it is now supported by virtually all web browsers[1] and major web servers over Transport Layer Security (TLS) using an Application-Layer Protocol Negotiation (ALPN) extension[1] where TLS 1.2 or newer is required.

HTTPv3 is the proposed successor to HTTP/2, and 2/3rd of web browser users (both on desktop and mobile) can already use HTTP/3, on the 17% of websites that already support it; uses UDP instead of TCP for the underlying transport protocol. Like HTTP/2, it does not obsolete previous major versions of the protocol. Support for HTTP/3 was added to Cloudflare and Google Chrome in September 2019 (since enabled by default), and can be enabled in the stable versions of Firefox and Safari.

3.List 5 difference between Browser js console vs Node js

- In browser “window” is a predefined global object which has functions and attributes, where as Nodejs doesn’t have it.
- In browser “location” is another predefined object, where as Nodejs doesn’t have it.
- In browser “require” is not predefined object, where as Nodejs has it.
- In browser module is not required, where as in Nodejs you have to keep your code inside the module.
- In browser “document” is a predefined object, where as Nodejs doesn’t have it.

4.what happens when you type a URL in the address bar in the browser

In general, when you type a URL into the browser location bar (and press enter) the browser needs to find server you are trying communicate with based on whatever you typed.