1.Difference between HTTP1.1 vs HTTP2.

1. HTTP/2 is much faster and more efficient than HTTP/1.1.One of the ways in which HTTP/2 is faster is in how it prioritizes content during the loading process.
2. In HTTP/2, developers have hands-on, detailed control over prioritization. This allows them to maximize perceived and actual page load speed to a degree that was not possible in HTTP/1.1
3. To speed up web performance, both HTTP/1.1 and HTTP/2 compress HTTP messages to make them smaller. However, HTTP/2 uses a more advanced compression method called HPACK that eliminates redundant information in HTTP header packets.
4. HTTP/1.1 loads resources one after the other, so if one resource cannot be loaded, it blocks all the other resources behind it. In contrast, HTTP/2 is able to use a single [TCP](https://www.cloudflare.com/en-in/learning/ddos/glossary/tcp-ip) connection to send multiple streams of data at once so that no one resource blocks any other resource.
5. HTTP version history

HTTP/0.9:

 HTTP/0.9 is extremely simple: requests consist of a single line and start with the only possible method GET followed by the path to the resource. There were no HTTP headers it can only transmit HTML files. There were no error codes in case of any problem .Instead a HTML file with the problem was sent back for human reference.

HTTP/1.0:

The HTTP headers has been introduced both for requests and response. With this the transmission of other documents other than HTML files were added with this version.

HTTP/1.1:

HTTP/1.1 was first published as RFC 2068 in January 1997.The first standardized protocol which was used for 15 years time duration.

A connection can be reused so this reduced the time to reopen it multiple times. Piping was added , so that a second request can be sent before the answer or response of first request.It lowered the latency of communication. HOST header was introduced which had the ability to host different domains in the same IP address.

HTTP/2.0:

Officially standardized, in May 2015, HTTP/2 has had much success. By July 2016, 8.7% of all Web sites were already using it, representing more than 68% of all requests. High-traffic Web sites showed the most rapid adoption, saving considerably on data transfer overheads and subsequent budgets.This rapid adoption rate was likely as HTTP/2 does not require adaptation of Web sites and applications.It is a binary protocol and also compresses headers and allows a server to populate data in a client cache. Page load speed improvements.

3.List 5 differences between Browser JS vs Node Js.

|  |  |
| --- | --- |
| Browser Javascript | Node.Js |
| Javascript can only run in browser. | Node js can run outside the browser. |
| It is used in client side. | It is used in server side. |
| It is upgraded version of ECMA script that uses Chrome’s V8 engine written in C++. | Nodejs is written in C,C++ and javascript |
| It is used in frontend development. | Nodejs is used in server side development. |
| It as capacity to add HTML tags. | It doesnt have capability to add HTML tags. |

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

URL-- Uniform Resource Locator is the address which you can find in the address bar of your web browser. It is a reference to a resource on the internet.

DNS-- Domain Name System maintains and maps the name of the website, i.e. URL, and particular IP address it links to. Every URL on the internet has a unique IP address which is of the computer which hosts the server of the website requested.

The steps below elaborates the process:

* Browser checks cache for DNS that we entered in order to find the corresponding IP address of website.It will loop through the below mentioned cache until the IP address is matched.

1. Browser Cache
2. Operating Systems Cache
3. Router Cache
4. ISP Cache

* If not found in cache, Internet Service Provider’s DNS server initiates a DNS query to find IP address of server that hosts the domain name.
* Browser initiates a TCP (Transfer Control Protocol) connection with the server using synchronize and acknowledge messages.
* Browser sends an HTTP request to the web server. GET or POST request.
* Server on the host computer handles that request and sends back a response. It assembles a response in some format like JSON, XML and HTML.
* Server sends out an HTTP response along with the status of response.
* Browser displays HTML content.