Task 01

Do a write up for the followings:?

**1**.*Difference between HTTP1.1 vs HTTP2*

|  |  |
| --- | --- |
| HTTP1.1 | HTTP2 |
| 1.It will send the resources one by  which may block other resources if one  of them blocked. | 1. It has a feature called multiplexing that allows it to send data streams all at once which do not block other resources. |
| 2. It does not have the server push option. | 2. It has server push option to push the content to client before client asks for it, in which client asks for several resources. |
| 3. To boost up the performance, it compresses  the messages in http. | 3. To boost up the performance, it uses the advanced technique HPACK which eliminates the redundant information from HTTP. |
| 4. Every Tcp requires a new control flow mechanism | 4. It requires only one tcp connection to implement a flow control mechanism |
| 5. The main thing that the loading speed of images may be slow and it will reflect only one page when we request . | 5. The main thing is that the images loads fast and it will reflect two pages when we request. |

**2**. *HTTP Version History*:

Hypertext Transfer Protocol was developed by Tim-Berners Lee. It is a application layer Protocol and it functions as a request-response model. It is a communication protocol of World Wide Web. It functions in client-server model. The initial version of http is HTTP/0.9 which do not transfer other type of files and can be terminated after the response, and it contains only get method and hypertext response. It is a single-line protocol. Http/1.0 is a expanded protocol with extended operations and officially recognised in the year 1996. It contains header files and can be terminated after the response like HTTP/0.9 and supports the methods of get, head, post. It is a browser friendly and cannot be limited to only response. Both HTTP/0.9 and Http/1.0 needs new connection for every request which in-turn opens TCP three way handshake. So, to improve the performance HTTP/1.1 is evolved.

HTTP/1.1 is a standardized protocol and contains the methods like get, post, put, delete, trace ,options. Every TCP requires a new control flow mechanism. To boost up the performance, it compress the messages in http and it does not have server push and multiplexing options. Since it does not have multiplexing option, it may block the other servers if one resource is failed to load. In this the speed of image loading is slow and reflect only one page when we request. To overcome this HTTP/2.0 is evolved with extended operations and it uses advanced option called HPACK to boost up the performance and deletes the redundant information from HTTP. It requires only one TCP connection to implement flow control mechanism. It has server push options to push the content to the client before the client asks where the request of client has several dozen resources. It has a feature of multiplexing which do not block the other resources if one resource got blocked and the speed of the image loading is fast. It uses weighted value to determine which resource loads first. Http/3.0 is a proposed successor of HTTP/2.0. It is only supported by some websites and uses UDP instead of TCP as a underlying communication protocol.

**3**. *List 5 difference between Browser JS(console) vs Nodejs*

|  |  |
| --- | --- |
| Browser JS | Node JS |
| 1. It is used to run on browser only | 1. It can be run outside the browser also. |
| 2. It is a client side and used in frontend development. | 2. It is a server side application and used in server side development. |
| 3. It is not headless and process response objects. | 3. It is headless and process request objects. |
| 4. Module is not necessary. | 4. Module is necessary. |
| 5. It can add html tags. | 5. It can't add html tags. |

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

1. The Browser checks the cache for a DNS(Domain Name System) record to find the corresponding IP address of the URL. DNS is a database that maintains the name of the website (URL) and the particular IP address it links to. Every single URL on the internet has a unique IP address assigned to it. The IP address belongs to the computer which hosts the server of the website we are requesting to access.

To find the DNS record, the browser checks four caches.

* First, it checks the browser cache. The browser maintains a repository of DNS records for a fixed duration for websites you have previously visited. So, it is the first place to run a DNS query.
* Second, the browser checks the OS cache. If it is not in the browser cache, the browser will make a system call (i.e., gethostname on Windows) to your underlying computer OS to fetch the record since the OS also maintains a cache of DNS records.
* Third, it checks the router cache. If it’s not on your computer, the browser will communicate with the router that maintains its’ own cache of DNS records.
* Fourth, it checks the ISP cache. If all steps fail, the browser will move on to the ISP. Your ISP maintains its own DNS server, which includes a cache of DNS records, which the browser would check with the last hope of finding your requested URL.

2. If the requested URL is not in the cache, ISP's DNS server initiates a DNS query to find the IP address of the server that hosts the website (URL).So, the domain name which you entered got converted into a DNS number.

3. The browser initiates the TCP connection with the server:

* As the client wants to establish a connection so it sends an SYN(Synchronize Sequence Number) to the server which informs the server that the client wants to start a communication.
* If the server is ready to accept connections and has open ports then it acknowledges the packet sent by the server with the SYN-ACK packet.
* In the last step, the client acknowledges the response of the server by sending an ACK packet. Hence, a reliable connection is established and data transmission can start now

4. The browser sends a http request to the server.

5. The server handles the incoming request and sends an HTTP response.

* 1xx: Informational: It means the request was received and the process is continuing.
* 2xx: Success: It means the action was successful.
* 3xx: Redirection: It means further action must be taken in order to complete the request. It may redirect the client to some other URL.
* 4xx: Client Error: It means some sort of error in the client’s part.
* 5xx: Server Error: It means there is some error on the server-side.

6.The browser displays the HTML content.

Now the browser gets the response and the HTML web page is rendered in phases. First, it gets the HTML structure and then it sends multiple GET requests to get the embedded links, images, CSS, javaScript files, etc and other stuff.