TASK 1:

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

|  |  |
| --- | --- |
| HTTP/1.1 | HTTP/2 |
| 1. HTTP/1.1, is the first standardized version of HTTP and was introduced in 1997. | 1. In 2015, the new version of HTTP was created which is called as HTTP/2. |
| 2. HTTP/1.1 loads resources one after the other, so if one resource cannot be loaded, it blocks all the other resources behind it. | 2. HTTP/2 is able to use a single TCP connection to send multiple streams of data at once so that no one resource blocks any other resource. |
| 3.HTTP/1.1 is slower and less efficient compare to HTTP/2. | 3. HTTP/2 is much faster because of how it prioritizes content during the loading process. |
| 4. In HTTP/1.1, developers do not have control over prioritization which makes it slower. | 4. 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. |
| 5. HTTP/1.1 uses Gizp compression. | 5. HTTP/2 uses a more advanced compression method called HPACK that eliminates redundant information in HTTP header packets. |

**Q.2 HTTP version history.**

**ANS: HTTP (**Hyper Test Transfer Protocol) is the underlying protocol of the World Wide Web. Developed by Tim Berners-Lee and his team. HTTP has evolved from an early protocol to exchange files in a semi-trusted laboratory environment, to the modern maze of the Internet.

Until now there are five protocols:

1. HTTP/0.9
2. HTTP/1.0
3. HTTP/1.1
4. HTTP/2
5. HTTP/3

**1. HTTP/0.9:** It is the initial version of HTTP. It is extremely simple.

* Request nature: single-line
* Methods supported: GET only
* Response type: hypertext only
* Connection nature: terminated immediately after the response
* No HTTP headers, No status/error codes, No URLs, No versioning

**2. HTTP/1.0:** It is a browser-friendly protocol.

* Provided header fields including rich metadata about both request and response (HTTP version number, status code, content type)
* Response: not limited to hypertext (Content type header provided ability to transmit files other than plain HTML files — e.g., scripts, stylesheets, media)
* Methods supported: GET, HEAD, POST.
* Connection nature: terminated immediately after the response

**3.HTTP/1.1:** This is the HTTP version currently in common use. Introduced critical performance optimizations and feature enhancements - persistent and pipelined connections, chunked transfers, compression/decompression, content negotiations, virtual hosting, faster response and great bandwidth savings by adding cache support.

* Methods supported:  GET, HEAD, PUT, POST, DELETE, TRACE, OPTIONS
* Connection nature: long-lived

**4. HTTP/2:**

* HTTP/2 is able to use a single TCP connection to send multiple streams of data at once so that no one resource blocks any other resource.
* HTTP/2 is much faster because of how it prioritizes content during the loading process.
* 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
* HTTP/2 uses a more advanced compression method called HPACK that eliminates redundant information in HTTP header packets.

**5.HTTP/3:** HTTP/3 provides a transport for HTTP semantics using QUIC transport protocol and an internal framing layer similar to HTTP /2.

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

|  |  |
| --- | --- |
| **JS** | **Nodejs** |
| **1.** JS is a programming language, which runs in web browsers. | **1.** Node.js is an interpreter or running environment for JavaScript, which holds a lot of requiring libraries and all. |
| **2.** JavaScript is normally used for any client-side activity for one web application. | **2.** Node JS mainly used for accessing or running any operating system for non-blocking operation. |
| **3.** JavaScript can run in any engine like Spider monkey (FireFox), JavaScript Core (Safari), V8 (Google Chrome). | **3.** Node JS only support the V8 engine, |
| **4.** JavaScript is normally following  Java programming language standard. | **4.** node JS is written in c++ and provides a V8 engine base browser avascript running engine, it helps us run a written javascript program in any browser environment. |
| **5.** Javascript is capable enough to add HTML and play with the DOM. | **5.**Nodejs does not have capability to add HTML tags. |

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

ANS: When you enter a URL into a web browser. The browser looks up the IP address for the domain named DNS. The browser sends a HTTP request to the server. The server sends back a HTTP response.

* Browser checks cache for DNS entry to find the corresponding IP address of website.  
  If cache is not found, then continues checking to the next until found.
  + Browser Cache
  + Operating Systems Cache
  + Router Cache
  + ISP Cache
* If not found in cache, ISP’s DNS server initiates a DNS query to find IP address of server that contains the domain name.
* Browser initiates a TCP connection with the server using synchronize (SYN) and acknowledge (ACK) 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, HTML etc.
* Server sends out an HTTP response along with the status of response.
* Browser displays HTML content.
* Finally we can see the website/ webpage.