1. Difference between HTTP1.1 vs HTTP2

HTTP1.1

Date of release: 1997

It supports connection reuse i.e. for every TCP connection there could be multiple requests and responses, and pipelining where the client can request several resources from the server at once. However, pipelining was hard to implement due to issues such as head-of-line blocking and was not a feasible solution.

It is relatively secure since it uses digest authentication, NTLM authentication.

Expands on the caching support by using additional headers like cache-control, conditional headers like If-Match and by using entity tags.

HTTP/1.1 provides faster delivery of web pages and reduces web traffic as compared to HTTP/1.0. However, TCP starts slowly and with domain sharding (resources can be downloaded simultaneously by using multiple domains), connection reuse and pipelining, there is an increased risk of network congestion.

HTTP2

Date of release: 2015

Uses multiplexing, where over a single TCP connection resources to be delivered are interleaved and arrive at the client almost at the same time. It is done using streams which can be prioritized, can have dependencies and individual flow control. It also provides a feature called server push that allows the server to send data that the client will need but has not yet requested.

It is relatively secure since it uses digest authentication, NTLM authentication.

HTTP/2 does not change much in terms of caching. With the server push feature if the client finds the resources are already present in the cache, it can cancel the pushed stream.

HTTP/2 utilizes multiplexing and server push to effectively reduce the page load time by a greater margin along with being less sensitive to network delays.

2. http version history

HTTP/0.9 The one-line protocol

There were no HTTP headers,only HTML files could be transmitted

HTTP/1.0 Building extensibility

HTTP headers has been introduced, both for the requests and the responses, allowing metadata to be transmitted and making the protocol extremely flexible and extensible.
With the help of the new HTTP headers, the ability to transmit other documents than plain HTML files has been added (thanks to the Content-Type header).

HTTP/1.1 The standardized protocol

A connection can be reused, saving the time to reopen it numerous times to display the resources embedded into the single original document retrieved.
Pipelining has been added, allowing to send a second request before the answer for the first one is fully transmitted, lowering the latency of the communication.
Chunked responses are now also supported.
Additional cache control mechanisms have been introduced.
Content negotiation, including language, encoding, or type, has been introduced, and allows a client and a server to agree on the most adequate content to exchange.
Thanks to the Host header, the ability to host different domains at the same IP address now allows server colocation.

HTTP/2 A protocol for greater performance

Over the years, Web pages have become much more complex, even becoming applications in their own right. The amount of visual media displayed, the volume and size of scripts adding interactivity, has also increased: much more data is transmitted over significantly more HTTP requests.

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

Browser	Nodejs
• "window" is a predefined global object which has functions and attributes, that have to deal with window that has been drawn.	• Nodejs doesn't have a predefined "window" object cause it doesn't have a window to draw anything.
• "location" is another predefined object in browsers, that has all the information about the url we have loaded.	• "location" object is related to a particular url; that means it is for page specific. So, node doesn't require that.
• "document", which is also another predefined global variable in browsers, has the html which is rendered.	• Ofcourse Nodejs doesn't have "document" object also, cause it never have to render anything in a page.
• Browsers may have an object named "global", but it will be the exact one as "window".	• Nodejs has "global", which is a predefined global object. It contains several functions that are not available in browsers, cause they are needed for server side works only.
• Browsers don't have "require" predefined. You may include it in your app for asynchronous file loading.	• "require" object is predefined in Nodejs which is used to include modules in the app.

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

When we enter a url in the address bar its job is to get converted into IP address which is linked to that URL for that The first place the computer looks is its local DNS cache, which stores DNS information that the computer has recently retrieved. If the records are not stored locally, the computer queries (or contacts) the ISP's recursive DNS servers. These machines perform the legwork of DNS queries on behalf of their customers. The recursive DNS servers have their own caches, which they check before continuing with the query.