Question No: 01

1. http1 loads the webpage very slower. It loads the webpage one by one. In this any interrupt occurred it automatically affects the other resources. But in http2 one plug in is used to connect all the resources so the interrupt will the affect all the resources.
2. In general the servers only serves the service when the client asks for it. In some cases there are any problems to serve with the device the user cannot ask the server to serve for thousand times. So that in http2 the server automatically serve the device without the user asked for it.
3. Small files load quickly than a larger one. So both the http1 and http2 are decided to compress the http. However the http2 uses the modern compression technique to compress the files so that the webpages are loaded very faster.

Question No:02

In 1989, while he was working at CERN, Tim Berners-Lee wrote a proposal to build a hypertext system over the Internet. Initially calling it the *Mesh*, it was later renamed to *World Wide Web* during its implementation in 1990. Built over the existing TCP and IP protocols, it consisted of 4 building blocks:

* A textual format to represent hypertext document *.*
* A simple protocol to exchange these documents, the *HypertText Transfer Protocol*(HTTP).
* A client to display (and accidentally edit) these documents, the first Web browser called *WorldWideWeb*.
* A server to give access to the document, an early version of *httpd.*

Question No: 03

1. Both in browser and node the javascript is the programming language which is used.
2. As the developer who uses the javascript knows how tough it will be to do all this in the browser. So that the node.js comes to play the very big advantage. So that we can learn bothe frontend and backend in the single language.
3. Building apps that run in the browser is completely different thing than the apps that run in the node.js
4. Since JavaScript moves so fast, but browsers can be a bit slow and users a bit slow to upgrade, sometimes on the web, you are stuck to use older JavaScript / ECMAScript releases.
5. You can use Babel to transform your code to be ES5-compatible before shipping it to the browser, but in Node, you won’t need that.

Question 04:

1. 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.
2. 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.
3. 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.
4. 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.