JS:

* Difference between MPA and SPA, when a url is hit on browser what no. of request happens between client and server at first.
* Promises / async and await / callbacks
* ES6 features
* Closure
* Hoisting
* What is the difference between normal function and arrow function?
* Why do we need promises? What are the problems that promise api resolves.
* difference between let and var
* Scopes in JS
* What is callback hell and how to avoid it.
* What are Mutable and immutable Data types in JS.
* Difference b/w session storage, local storage and cookies.
* How will you mutate objects in javascript, and why.
* What data types passed by value or by reference.
* How many types of design patterns are there in javascript.
* ES6 class vs constructor function
* remove duplicates in array using es6
* ES6 map vs forEach

NodeJs:

* What is nodejs? Where and Why we can use Nodejs?
* What's an Event loop in node js ?
* What's blocking and non blocking in nodejs ?
* What's the difference b/w node js and others ?
* EventListener and EventEmitter in nodejs?
* Modules in nodejs?
* Streams in nodejs?

JS Frameworks:

* Why React when we have Angular or vise versa ?
* What's React doing which Angular wasn’t able to do in Angular 1.x ?
* Why react is a library and Angular is a framework.
* Ways to store states in react
* How react lifecycle works
* Jsx vs JS
* How to pass Props to siblings or child component
* Dumb component vs smart component
* Higher order components
* Immutability in React
* Redux Flow
* React Hooks
* Memoization
* what is Virtual DOM in React
* Reconciliation and diffing algorithm in React
* How React / Angular change detection cycle works
* What is an observable
* Observable vs Promise
* Which design pattern redux uses. (singleton)

-- HOW A WEB LOADS?

The browser goes to the DNS server, and finds the real address(IP) of the server that the website lives on (you find the address of the shop).

The browser sends an HTTP request message to the server, asking it to send a copy of the website to the client (you go to the shop and order your goods). This message, and all other data sent between the client and the server, is sent across your internet connection using TCP/IP.

The browser parses the HTML file first, and that leads to the browser recognizing any <link>-element references to external CSS stylesheets and any <script>-element references to scripts.

As the browser parses the HTML, it sends requests back to the server for any CSS files it has found from <link> elements, and any JavaScript files it has found from <script> elements, and from those, then parses the CSS and JavaScript.

The browser generates an in-memory DOM tree from the parsed HTML, generates an in-memory CSSOM structure from the parsed CSS, and compiles and executes the parsed JavaScript.

As the browser builds the DOM tree and applies the styles from the CSSOM tree and executes the JavaScript, a visual representation of the page is painted to the screen, and the user sees the page content and can begin to interact with it.