

- **Javascript**

- What is your favourite feature of ES6?
- What is the [difference between var, let, and const](#)?
- What is callback hell? How do you avoid it?
- [What is a promise?](#)
- [What is a closure?](#) (See end of document for closure example)
- What is async and await keywords? How do you use them? Write an example.
- What is a pure function?
- What is the [event loop](#)?
- Explain "hoisting".
- Describe event bubbling. **
- Can you describe the main difference between the Array.forEach() loop and Array.map() methods and why you would pick one versus the other?
- What's the difference between a variable that is: null, undefined or undeclared?
- Explain event delegation.
- Explain how *this* works in JavaScript.
 - Can you give an example of one of the ways that working with *this* has changed in ES6?
- What are the benefits of an arrow function? What's the difference between arrow function and non arrow function?
- What is the difference between == and ===?
- What is the definition of a higher-order function?
- Can you give an example for destructuring an object or an array?
- How can you share code between files?(using diff modules) **
- What is the difference of undefined and null?
- What is Class expression?
 - What is prototype-based inheritance?
- <https://github.com/getify/You-Dont-Know-JS>

- **React**

- What is the lifecycle of React components? What happens at different steps in the lifecycle?
 - This article has a [good answer](#) and a few other questions
- What is the difference between props and state?
- When should you use props?
- How do you pass data through components? ([Link 1](#), [Link 2](#))
- How do you avoid full component re-renders?
- [Functional vs. Class components](#)
- What is a PureComponent?
- What is an HOC?
- Newer React Features: What are hooks? Why use them?
- When should you use Redux? What is Redux?
- For each of the lifecycle methods, how do you achieve the same result with functional components

- How do you do componentWillUnmount with functional components
 - What is useRef used for?
 - What is useCallback, React.memo, and how does it relate to PureComponent?
 - How does React compare with other frameworks (Vue.js, Angular)?
 - What are different ways of [state management with React](#)?
 - You can see more questions [here](#)
- **CSS**
 - What is bootstrap? Bootstrap grid? Write some code in vanilla CSS or use bootstrap to achieve responsive layout (media queries, etc.)
 - [Pseudo-classes](#)
 - [Pseudo-elements](#)
 - CSS Box Model, display types, etc.
 - What does the following piece of css do?
 - CSS specificity
 - CSS Grid, CSS Flexbox
 - Difference between an ID and class selector?
- **Web development**
 - Check out this blog post on [general web development topics](#)
 - What is HTTP? What is DNS?
 - What is CORS? How does it work?
 - What happens when you go to a webpage and hit enter?
 - What specifically happens on the front-end?
 - What happens on the back-end?
 - How do companies scale websites?
 - Load balancers
 - Caching
 - Horizontal vs. vertical scaling
 - Microservice vs. monolithic architecture
 - How do you optimize the performance of a website? On the frontend? On the backend?
 - Caching - front-end, back-end, database
 - CDNs
 - Cache headers
 - Image optimization, minification, bundling, etc.
 - How do you debug front-end applications in the browser?
 - How do you debug back-end applications?
 - What is a scraper vs. a crawler? What are general techniques to crawling a website?
 - What is webpack?
 - What does a webpack config file do?
 - What is Babel?
- **Node**
 - Write a basic express route

- What is `app.post(...)` etc.
 - How to get params vs. body, etc.
 - Difference between post, get, delete, put, etc.
- [High level questions](#)
- Here are some [good questions](#)
- [Is NodeJS single-threaded?](#)
- Here are a couple of good questions about [Node](#)
 - You will want to know what Node is, why it is single threaded, etc.
- Here are a few questions you should know about [Express](#)
- **Security**
 - How does HTTPS work? What is the difference between HTTP and HTTPS?
 - Handshake protocol for HTTPS
 - What are methods of securing a website and preventing attacks?
 - How do you handle authentication? How do companies restrict access to certain pages?
 - How can you write scrapers to circumvent user authentication
 - How would you log in as a user using a scraper?
 - What are cookies vs. local storage vs. session storage? What are their differences?
 - What is a CSRF token?
 - What is JWT? How does it work, what does it do?
- **Back-end / Databases**
 - What is a RESTful API? What are the benefits of a RESTful API?
 - What is RESTful naming conventions?
 - What are the different HTTP methods (GET, POST, PUT, DELETE, etc.) and what are they used for
 - What are the expected status codes for HTTP methods
 - Describe CRUD operations
 - What is the difference between SOAP and RESTful
 - SQL vs. NoSQL ([Check out this presentation on database design](#))
 - How do you make relationships in SQL? How do those relationships look different in NoSQL? (MongoDB has some good [resources](#) on that)
 - How would you design a Blog or a Reddit type application using MongoDB? How would you design dropbox, or google maps, or a social media site?
 - Design a URL shortener application
 - What is Mongoose?
 - Practice writing some simple MongoDB or SQL queries (start easy and go to harder ones like mapreduce)
 - See our interview prep (<https://hatchways.io/app/#/dashboard/prep>) for system design question and a comments table question on our prep
- **Testing**
 - Check out [this presentation of testing](#)
 - What are the differences between integration, unit and E2E testing

- How do testing tools relate to scraping? How can they be used (especially E2E tools)?
 - What is mocking? What is stubbing?
 - What is test coverage?
 - What is linting?
- What is selenium? What is phantomJS?
 - How can you use these tools for scraping?
 - What is a User Agent? How does that relate to scraping?
 - If a company is rendering data with javascript (like onscroll listeners, etc.), how would you scrape that content?
- **CI / CD / Docker**
 - What is Docker? What does it do and how does it help?
 - What is CI? What is CD?
 - What is CircleCI or Jenkins? How does it work?