Frontend Interview Questions and Answers

Codingkite

Hey there, I'm Rohit
Admin of Codingkite

Codingkite

"I've made some valuable notes for you from three front-end interviews my friend recently aced. They're all yours for free!"

Html & Css

1. What is HTML?

- HTML stands for HyperText Markup Language. It is the standard markup language for creating web pages and web applications.

2. What is CSS?

- CSS stands for Cascading Style Sheets. It is a style sheet language used for describing the presentation of a document written in HTML.

3. What is the difference between HTML and CSS?

- HTML is used to structure content on a web page, while CSS is used to style and layout the content.

4. What are the different types of selectors in CSS?

- CSS selectors are patterns used to select the elements you want to style. Some common types of selectors include element selectors, class selectors, ID selectors, attribute selectors, and pseudo-selectors.

5. Explain the box model in CSS.

- The CSS box model describes the rectangular boxes that are generated for elements in the document tree. It consists of content, padding, border, and margin. The content area contains the actual content of the element, while padding is the space between the content and the border, border is the border around the padding, and margin is the space outside the border.

6. What is the difference between padding and margin in CSS?

- Padding is the space between the content of an element and its border, while margin is the space outside the border of an element.

7. What is the difference between inline and block elements in HTML?

- Block-level elements start on a new line and take up the full width available, while inline elements do not start on a new line and only take up as much width as necessary.

8. What is the importance of the 'float' property in CSS?

- The 'float' property in CSS is used to move an element to the left or right, allowing other elements to wrap around it. It is commonly used for creating layouts, such as floating a sidebar next to the main content.

9. What is the 'display' property in CSS used for?

- The 'display' property in CSS is used to specify the type of rendering box used for an element. It can change an element's behavior from block-level to inline or vice versa, among other options like flex or grid.

10. How do you center an element horizontally and vertically in CSS?

- To center an element horizontally, you can set its margin-left and margin-right properties to 'auto' and its width to a specific value. To center an element vertically, you can use the flexbox or grid layout, or use the transform property with translateY(-50%) along with absolute positioning.

JavaScript

1. What is JavaScript?

- JavaScript is a high-level, interpreted programming language used to make web pages interactive and dynamic. It is primarily used for client-side scripting but can also be used for server-side development (Node.js).

2. What are the different data types in JavaScript?

- JavaScript has several data types including:
- Primitive data types: number, string, boolean, null, undefined, symbol (added in ES6)
- Reference data types: object, array, function

3. What is hoisting in JavaScript?

- Hoisting is a JavaScript mechanism where variables and function declarations are moved to the top of their containing scope during the compilation phase. This means that you can use variables and functions before they are declared.

4. What is closure in JavaScript?

- A closure is the combination of a function and the lexical environment within which that function was declared. Closures allow functions to retain access to variables from their containing scope even after the scope has closed.

5. What is the difference between == and === operators in JavaScript?

- The == operator checks for equality after performing type coercion, while the === operator checks for equality without type coercion. In other words, === checks both value and type equality, whereas == only checks value equality after converting both operands to a common type.

6. What are the different ways to define a function in JavaScript?

- Functions in JavaScript can be defined using function declarations, function expressions, arrow functions (added in ES6), and methods (functions defined within objects).

7. What is event bubbling and event capturing in JavaScript?

- Event bubbling and event capturing are two mechanisms for handling events in the DOM. Event bubbling occurs when an event starts at the innermost element and works its way up to the outermost element. Event capturing, on the other hand, occurs when the event is captured from the outermost element and works its way down to the innermost element.

8. Explain the concept of prototypal inheritance in JavaScript.

- Prototypal inheritance is a way of creating objects in JavaScript where objects inherit properties and methods from other objects (their prototypes). Each object in JavaScript has a prototype chain which it follows to inherit properties and methods from its prototype.

9. What is the purpose of the 'this' keyword in JavaScript?

- The 'this' keyword in JavaScript refers to the object that is currently executing the function. Its value is determined by how a function is called, and it allows access to the object's properties and methods within that function.

10. What are promises in JavaScript?

- Promises are objects used to represent the eventual completion (or failure) of an asynchronous operation and its resulting value. They provide a cleaner way to handle asynchronous operations compared to callbacks, allowing for better error handling and chaining of multiple asynchronous operations.

React.js

1. What is React.js?

- React.js is an open-source JavaScript library for building user interfaces, developed by Facebook. It allows developers to create reusable UI components and efficiently manage the state of the application.

2. What are the key features of React.js?

- Some key features of React.js include:
  - Virtual DOM for efficient updates
  - Component-based architecture
  - JSX syntax for defining UI components
  - Unidirectional data flow
  - Reusable and composable UI components
  - React Hooks for managing state and side effects (introduced in React 16.8)

3. What is JSX?

- JSX (JavaScript XML) is a syntax extension for JavaScript that allows developers to write HTML-like code within JavaScript. It simplifies the process of creating React elements and makes the code more readable.

4. What is a component in React?

- A component in React is a reusable UI element that encapsulates a piece of UI functionality. Components can be composed together to build complex UIs, and they can have their own state and lifecycle methods.

5. What is state in React?

- State in React is a JavaScript object that represents the current data of a component. It determines how a component renders and behaves. State can be modified using the `setState()` method, triggering re-rendering of the component.

6. What are props in React?

- Props (short for properties) are inputs to React components that are passed from parent to child components. They are read-only and help to make components reusable by providing a way to customize their behavior and appearance.

7. What is the difference between state and props in React?

- State is managed internally by a component and can be changed over time, whereas props are passed to a component from its parent and cannot be changed within the component.

8. What are React Hooks?

- React Hooks are functions that allow developers to use state and other React features without writing a class. They were introduced in React 16.8 and provide a more concise way to manage state and side effects in functional components.

9. What are the different lifecycle methods in React?

- In class components, React provides several lifecycle methods such as `componentDidMount()`, `componentDidUpdate()`, and `componentWillUnmount()` that allow developers to perform actions at different stages of a component's lifecycle.

10. What is the significance of keys in React lists?

- Keys are special string attributes used by React to identify which items in a list have changed, been added, or been removed. They help React optimize the rendering process by minimizing re-renders and improving performance.

11. What is the purpose of useEffect() Hook in React?

- The `useEffect()` Hook in React is used to perform side effects in functional components. It replaces lifecycle methods like `componentDidMount()`, `componentDidUpdate()`, and `componentWillUnmount()` in class components.

12. What are controlled components in React?

- Controlled components in React are form elements whose value is controlled by React state. This means that the value of the form element is derived from the component's state, and any changes to the value are handled by updating the state.

13. What is the Context API in React?

- The Context API in React provides a way to pass data through the component tree without having to pass props down manually at every level. It allows components to subscribe to a context and access the data it provides.

14. What is React Router?

- React Router is a popular library for handling routing in React applications. It provides a declarative way to define routes and navigate between different views or pages in a single-page application.

15. What are Higher Order Components (HOCs) in React?

- Higher Order Components (HOCs) are a pattern in React where a component takes another component as an input and returns a new component with enhanced functionality. HOCs are often used for code reuse, logic abstraction, and cross-cutting concerns like authentication and authorization.

16. What are React Fragments?

- React Fragments are a feature in React that allows developers to group multiple elements without adding extra nodes to the DOM. They provide a way to return multiple elements from a component's render method without wrapping them in a single parent element.

17. What are the differences between class components and functional components in React?

- Class components are ES6 classes that extend the `React.Component` class and use a render method to return React elements. Functional components are JavaScript functions that return React elements directly. With the introduction of Hooks in React 16.8, functional components can now also manage state and have lifecycle methods, making them more similar to class components.

18. What is server-side rendering (SSR) in React?

- Server-side rendering (SSR) in React refers to the process of rendering React components on the server and sending the generated HTML to the client. This can improve performance and SEO by providing faster initial page loads and better search engine indexing.

19. What is code splitting in React?

- Code splitting in React is a technique used to split the application's JavaScript bundle into smaller chunks, which are then loaded on-demand or in parallel. This helps reduce the initial load time of the application and improve performance, especially for larger applications.

20. How can you optimize performance in React applications?

- Performance optimization techniques in React include using shouldComponentUpdate or React.memo for optimizing re-renders, implementing code splitting to reduce bundle size and improve load times, lazy loading components or routes, optimizing network requests, and using React Profiler to identify performance bottlenecks.

Git, GitHub, Bootstrap, Tailwind CSS:

1. What is Git?

- Git is a distributed version control system used to track changes in files and coordinate work among multiple developers. It allows users to work on projects simultaneously, keep track of changes, and merge changes back into the main codebase.

2. What is GitHub?

- GitHub is a web-based platform that provides hosting for Git repositories. It offers features such as version control, collaboration tools, code review, issue tracking, and continuous integration.

3. What is Bootstrap?

- Bootstrap is a free and open-source front-end framework used to design and develop responsive and mobile-first websites and web applications. It provides a collection of pre-designed components, including buttons, forms, navigation bars, and more, along with CSS and JavaScript libraries for styling and interactivity.

4. What is Tailwind CSS?

- Tailwind CSS is a utility-first CSS framework that provides a set of pre-designed utility classes to style HTML elements. It allows developers to quickly build custom designs by applying utility classes directly to HTML elements, without having to write custom CSS styles.

5. What is CSS?

- CSS (Cascading Style Sheets) is a style sheet language used to describe the presentation of a document written in HTML. It allows developers to control the layout, appearance, and formatting of HTML elements on a web page.

6. What is the difference between Git and GitHub?

- Git is a version control system, whereas GitHub is a web-based platform for hosting Git repositories and collaborating on projects. Git is the tool used for version control, while GitHub provides additional features such as issue tracking, code review, and collaboration tools.

7. How do you create a new branch in Git?

- To create a new branch in Git, you can use the `git checkout -b <branch_name>` command. This command creates a new branch with the specified name and switches to it.

8. What is a pull request in GitHub?

- A pull request in GitHub is a way to propose changes to a repository and request feedback from other contributors. It allows developers to submit their changes for review and merge them into the main codebase after approval.

9. How do you add files to a Git repository?

- To add files to a Git repository, you can use the `git add <file>` command to stage the changes, followed by the `git commit -m "commit message"` command to commit the changes to the repository.

10. What is the purpose of media queries in CSS?

- Media queries in CSS are used to apply different styles to a web page based on the characteristics of the device or viewport, such as screen size, resolution, orientation, and more. They allow developers to create responsive designs that adapt to different devices and screen sizes.

Check My Ebook - https://topmate.io/codingkite/700886

My website - https://codingkite.com/