

React Interview

1. React Basic

1.1 What is React.js?

React.js is a front-end and open-source library for building complex and reusable components. The components can run in website and mobile applications.

Important Feature

1. Server-Side Render
2. Virtual DOM
3. Unidirectional data binding
4. JSX

1.2 What is React Hooks?

Before React 16.8, only class components were used for state management and lifecycle method, function components were only used for stateless components. After React 16.8, React has hooks and allows function components to manage state and lifecycle, which make coding easier, cleaner and more flexibility.

There are two rules.

Only function components can use hooks.

React Hooks **must be called only at the top level**. It is not allowed to call them inside the nested functions, loops, or conditions.

1.3 What is useState()

useState is one of hooks in React. It will return a tuple, the first param is the current state, the second param is a function which allow us to update the state. It helps us control the variable state. For example, I want to develop a counting website. Every time I click the button, the count will increase. I can use useState return [count, setCount], count is the value, setCount is the function that helps us increase.

1.4 What is useEffect()

useEffect is a hooks for processing side effects. We can use effect to do something after rendering and state changed.

useEffect() accepts two params. **useEffect(callback,[dependencies])**

Callback function will execute immediately when page renders first time and dependencies changed every times. When dependencies are empty array. The callback will only excute once the first time.

1.5 What is JSX.

JavaScript XML. It allows us write HTML code inside the JS code without using function

1.6 What are keys in React?

We can use keys for list rendering by Arrays.map. Give every item a unique key. Keys should be from API data.

1.7 What is virtual DOM

Every real DOM has copied virtual DOM, which has the same properties. And virtual DOM has two virtual DOM, one is prev virtual DOM, one is updated virtual DOM. And React finds the diff between two virtual DOM, and rendering the changed real DOM in the website, instead of rendering all DOM. (Because most JavaScript frameworks update the entire DOM even when a small part of the DOM changes.)

Two virtual DOM data structures: tree, only same level tag will compare

1.9 What is the difference between props and state.

| Props | State |
|-----------------------------------|---|
| Immutable | Owned by its component |
| Has better performance | Locally scoped |
| Can be passed to child components | Writeable/Mutable |
| | has <code>setState()</code> method to modify properties |
| | Changes to state can be asynchronous |
| | can only be passed as props |

1.10 What is the difference between hooks and classes.

| React Hooks | Classes |
|--|---|
| It is used in functional components of React. | It is used in class-based components of React. |
| It will not require a declaration of any kind of constructor. | It is necessary to declare the constructor inside the class component. |
| It does not require the use of <code>this</code> keyword in state declaration or modification. | Keyword <code>this</code> will be used in state declaration (<code>this.state</code>) and in modification (<code>this.setState()</code>). |
| It is easier to use because of the <code>useState</code> functionality. | No specific function is available for helping us to access the state and its corresponding <code>setState</code> variable. |
| React Hooks can be helpful in implementing Redux and context API. | Because of the long setup of state declarations, class states are generally not preferred. |