

EP6- Exploring the world

Why we need states in React and not use normal variables?

In order to keep the ui layer in sync with the data layer, we cannot use normal variables because React will not be able to know that there is any change.

So what we do, we take a state variable, make changes to it using setState and react will be able to know that state has been changed.

Whenever there is a state change , react will rerender the entire component, compare the new virtual dom with old dom by diff algo, and will reflect those changes to real dom and will reflect on ui.

React only traces state variables . It does not care what other normal variables are doing.

Why we are not calling service call normally but use useEffect?

If there are nothing in dependency array in useEffect it will be called for every re-render

See if there is no dependency in useEffect, then it will be called in every render. And render you know happens when any state changes. So useEffect will be called if any state changes.

But if you add an empty dependency, then it will be called for the first render, it will not call on every state change , means it will not be called on every render.

Component is rendered first and then useEffect will be called.

If there is no dependency, then useEffect will be called on every render, and that happens on first time component is mounted and if any state of component changes.

If dependency is an empty array, then it will called when component is rendered for the first time.

And suppose dependency array is [abc], then useEffect will be called on first mount and when there is any change in abc state value.

First component will render and then use effect is called