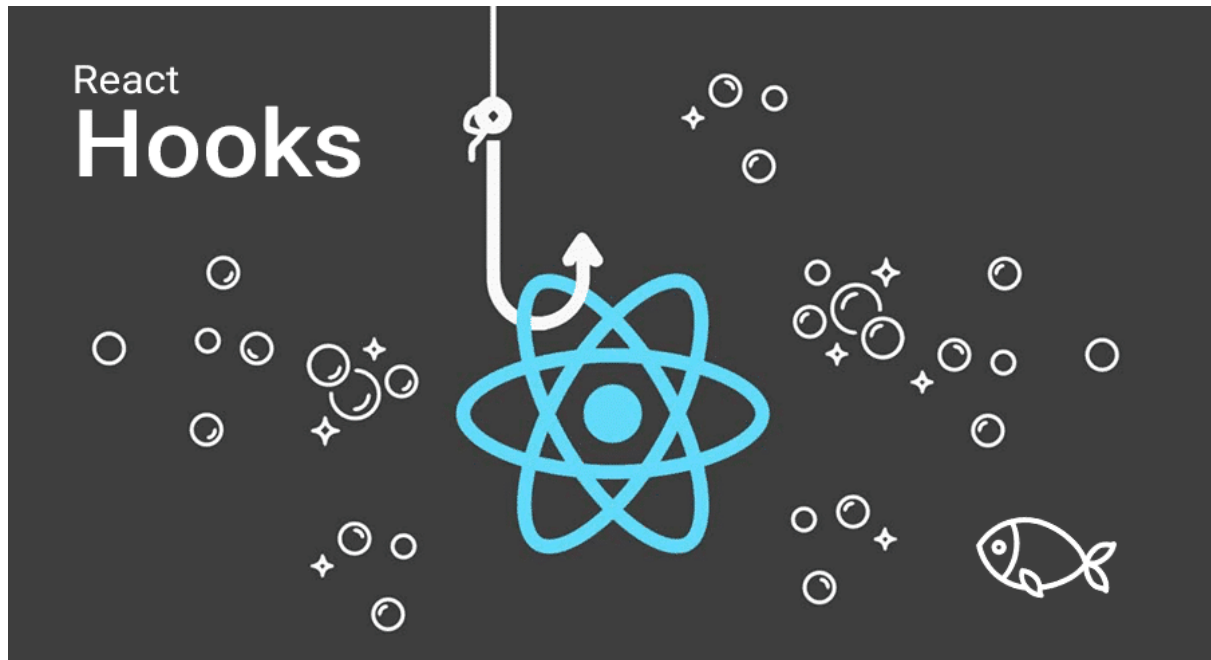


Interview Questions



Q1.What are hooks in React? (Gojek)

Answer: Hooks are a new feature added in React v16.8. It allows using all React features without writing class components. For example, before version 16.8, we need a class component to manage the state of a component. Now we can keep the state in a functional component using **useState** hook.

Q2.Why React hooks was introduced? (Amazon)

Answer: One reason to introduce hooks was the complexity in dealing with this keyword inside class components. If not handled properly, this will take some other value. That will result in breaking lines like this.setState() and other event handlers. Using hooks, we avoid that complexity when working with functional components.

Class components do not minify very well and also make hot reloading unreliable. That is another inspiration to bring hooks.

Another reason is that there is no specific way to reuse stateful component logic. Even though HOC and render props patterns address this problem, that asks for modifying the class component code. Hooks allow sharing stateful logic without changing the component hierarchy.

The fourth reason is, in a complex class component, related codes are scattered in different lifecycle methods. Example, in case of a data fetching, we do that mainly in `componentDidMount()` and `componentDidUpdate()`. Another example is, in the case of event listeners, we use `componentDidMount()` to bind an event and `componentWillUnmount()` to unbind. Hooks instead help to place related code together.

Q3.How useState hook works? What is/are the arguments accepted by this hook and what is returned by the hook? (OYO)

Answer:useState hook is a function that is used to store state value in a functional component. It accepts an argument as the initial value of the state. It returns an array with 2 elements. The first element is the current value of the state. The second element is a function to update the state.

We import useState first from React by

```
import React, { useState } from "react";
```

Later we use useState like:

```
const [currentStateValue, functionToUpdateState] = useState(initialStateValue);
```