React component

Earlier, the developers write than thousands of lines of code for developing a single page application. These application follow the traditional DOM structure, and making changes in them was a very challenging task. if any mistakes found, it manually searches the entire application and update accordingly.

The component based approach was introduced to overcome issues. In this approach the entire application is divided into a small logical group of code, which is known as components.

A component is considered as the core building blocks of a react application. It makes the task of building UI much easier. Each component exists in the same space, but they work independently from one another and merge all in a parent component, which will be the final UI of the application.

Every react component have their own structure, method as well as API. React component can be reused throughout our application, making it to easy to maintain and update.

In react we have mainly two types of components.

1. Functional components
2. Class components
3. Functional components

Functional components are used to define UI element and encapsulate behavior in a modular way. Functional components are essentially javascript functions that take props as a argument and return react elements to be rendered.

React component lifecycle methods

React class components have several lifecycle methods, which are executed at specific points during the components lifecycle. These methods are allow us to perform various actions, such as fetching data, updating the UI, or cleaning up resources.

Constructor

The constructor runs before the class component mounts. Typically, a constructor is only userd for two purpose in react.

**For declare state or**

**For bind class methods to the class instance**

Each component in react has a lifecycle which we can monitor and manipulate during its three main phases.

**Mounting**

It is initial phase of a component in this phase component is created and inserted into the dom. react has four built-in methods that gets called, when mounting a component

constructor ()

getDerivedStateFromProps ()

render ()

componentDidMount ()

constructor ()

The constructor is a method used to initialized object state in a class. It automatically called during the creation of object in a class.

The constructor in a react component is called before the component is mounted.

static getDerivedStateFromProps(props,state)

it is a static method and call just before render() method in both mounting and updating phase. It used when the state of a component depends on changes of props.

componentDidMount()

it is allowed to execute the react code when the component is already in the dom . all ajax request and the dom or state updation should be implemented in the componentDidMount() method block. We can also set up all the major subscription here but to avoid any performance issues, always remember to unsubscribe them in componentWillUnmount().

**Updating**

The updating phase in a react component lifecycle occurs when the component is already in the dom, ant it’s being re-rendered due to changes in its state or props.

static getDerivedStateFromProps(props, state)

shouldComponentUpdate(nextProps, nextState)

render()

getSnapshotBeforeUpdate(prevProps, prevState)

componentDidUpdate(prevProps, prevState, snapshot)

**Unmounting**

The unmounting phase is the final phase in a react components lifecycle, and it occurs when the component is remove from the dom, it involves cleanup tasks and is crucial for freeing up resources and preventing memory leaks.