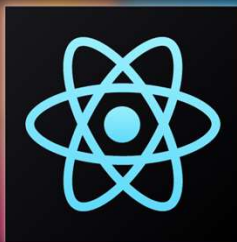




Venkatesh Mogili
#WebGuru

తెలుగు లో



What's New In? React 18



Contents



- What's New In React 18?
- How to Upgrade to React 18?
- What are the new hooks introduced in React 18?



What's New In React 18?



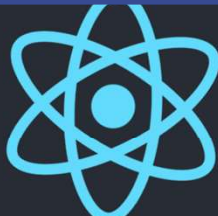
- **Rendering APIs**
- **StrictMode**
- **Automatic Batching**
- **Transitions**
- **New Suspense Features**
- **New Hooks**



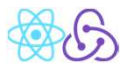
How To Upgrade To React 18?



```
npm install react@18.1.0 react-dom@18.1.0
```



```
npx create-react-app project-name
```



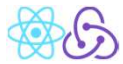
Rendering APIs and StrictMode



- **createRoot**
- **unmount()**
- **<StrictMode>**

ReactDOM.render is no longer supported in React 18.

If your app doesn't work after upgrading, check whether it's wrapped in <StrictMode>.



Automatic Batching



- React 18 adds **out-of-the-box performance improvements** by doing more **batching by default**.
- **Batching is** when **React groups multiple state updates into a single re-render** for better performance.
- **Before React 18**, we only **batched updates inside React event handlers**. Updates inside of **promises**, **setTimeout**, **native event handlers**, or any other event were **not batched in React by default**.
- **Starting in React 18 with createRoot**, all updates will be **automatically batched**, no matter where they originate from.

```
// Before React 18 only React events were batched

function handleClick() {
  setCount(c => c + 1);
  setFlag(f => !f);
  // React will only re-render once at the end (that's batching!)
}

setTimeout(() => {
  setCount(c => c + 1);
  setFlag(f => !f);
  // React will render twice, once for each state update (no batching)
}, 1000);
```

```
import { flushSync } from 'react-dom';

function handleClick() {
  flushSync(() => {
    setCounter(c => c + 1);
  });
  // React has updated the DOM by now
  flushSync(() => {
    setFlag(f => !f);
  });
  // React has updated the DOM by now
}
```

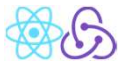


Transitions



- A transition is a new concept in React to distinguish between **urgent and non-urgent updates**.
- **Urgent updates** reflect direct interaction, like typing, clicking, pressing, and so on.
- **Transition updates** transition the UI from one view to another.
- **useTransition** – hook used for non urgent updates.
- **[isPending, startTransition]**

```
function App() {  
  const [isPending, startTransition] = useTransition();  
  const [count, setCount] = useState(0);  
  
  function handleClick() {  
    startTransition(() => {  
      setCount(c => c + 1);  
    })  
  }  
  
  return (  
    <div>  
      {isPending && <Spinner />}  
      <button onClick={handleClick}>{count}</button>  
    </div>  
  );  
}
```



New Suspense Features



- Suspense lets you declaratively specify the **loading state** for a part of the component tree if it's not yet ready to be displayed.

```
<Suspense fallback={<Spinner />}>  
  <Comments />  
</Suspense>
```



What's Are The New Hooks In React 18?



- **useTransition, startTransition**
- **useId**
- **useDeferredValue**
- **useSyncExternalStore**
- **useInsertionEffect**



Summary



- ✓ **What's New In React 18?**
- ✓ **How to Upgrade to React 18?**
- ✓ **What are the new hooks introduced in React 18?**

Thank You For Watching



Quiz



Tasks



Experiment

<https://vmtraining.netlify.app>