

What Are React Hooks?

- Hooks are functions that let you "hook into" React's state and lifecycle features in functional components.
- **Why Hooks?** Hooks eliminate the need for class components, simplifying code structure and making functional components more powerful.
- **Types of built in hooks:**
 - state hooks
 - context hooks
 - ref hooks
 - performance hooks
 - resource hooks



These are some common used hooks :

- **useState:**
- Program to demonstrate the use of a state variable in a functional component and how to update it accordingly.

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

function App() {
  const [click, setClick] = useState(0);
  // using array destructuring here
  // to assign initial value 0
  // to click and a reference to the function
  // that updates click to setClick
  return (
    <div>
      <p>You clicked {click} times</p>

      <button onClick={() => setClick(click + 1)}>
        Click me
      </button>
    </div>
  );
}

export default App;
```

- **useEffect:**
- The `useEffect` in ReactJS is used to handle the side effects such as fetching data and updating DOM.

```
import React, { useMemo, useState, useEffect } from "react"
import { getPatient, deletePatient } from "../../service/api"

const [alluserlist, setallusers] = useState({ users: [] })
const [loading, setLoading] = useState(true);

useEffect(() => {
  fetchPatientData()
}, [])

const fetchPatientData = async () => {
  try {
    let response = await getPatient()
    if (response.error) {
      console.log("Error while fetching data:", response.error)
    } else {
      setallusers({ users: response.data || [] })
    }
  } catch (error) {
    console.log("Error while fetching data:", error)
  } finally {
    setLoading(false);
  }
}
```

- **useParams:**
- The useParams hook returns an object of key/value pairs of the dynamic params from the current URL that were matched by the <Route path>. Child routes inherit all params from their parent routes.

```
import * as React from 'react';
import { Routes, Route, useParams } from 'react-router-dom';

function ProfilePage() {
  // Get the userId param from the URL.
  let { userId } = useParams();
  // ...
}

function App() {
  return (
    <Routes>
      <Route path="users">
        <Route path="me" element={...} />
        <Route path=":userId" element={<ProfilePage />} />
      </Route>
    </Routes>
  );
}
```