|  |
| --- |
| **API Test Cases** |
| 1. **async/await**   import {useEffect, useState} from 'react';  import './App.css';  function App() {  const [data, setdata] = useState([]);  useEffect(()=>{  const fetchUsers = async () => {  const response = await fetch('https://jsonplaceholder.typicode.com/todos/');  console.log(response);  const data = await response.json();  setdata(data);  };  fetchUsers();  },[])  return (  <div className="App">  <h1>Understand Fetching Data from Server !</h1>  <br />  <div>  {data.map(item => (  <li key={item.id}>  <p>Title: {item.title}</p>  <p>Completed: {item.completed ? 'Yes' : 'No'}</p>  </li>  ))}  </div>  </div>  );  }  export default App;  Output:  A screenshot of a computer  Description automatically generated |
| 1. promise   import {useEffect, useState} from 'react';  import './App.css';  function App() {  const [data, setdata] = useState([]);  useEffect(()=>{  fetch('https://jsonplaceholder.typicode.com/todos/')  .then((res)=>  res.json()  )  .then((data)=>  setdata(data)  ).catch((err)=>console.log(err));    },[])  return (  <div className="App">  <h1>Understand Fetching Data from Server !</h1>  <br />  <div>  {data.map(item => (  <li key={item.id}>  <p>Title: {item.title}</p>  <p>Completed: {item.completed ? 'Yes' : 'No'}</p>  </li>  ))}  </div>  </div>  );  }  export default App;  Same Output:  A screenshot of a computer  Description automatically generated |
| 3.  import {useEffect, useState} from 'react';  import './App.css';  function App() {  const [data, setdata] = useState([]);  useEffect(()=>{  fetch('https://jsonplaceholder.typicode.com/todos/')  .then((res)=>{  return res.json()  })  .then((data)=>{  return setdata(data)  }).catch((err)=>console.log(err));    },[])  return (  <div className="App">  <h1>Understand Fetching Data from Server !</h1>  <br />  <div>  {data.map(item => (  <li key={item.id}>  <p>Title: {item.title}</p>  <p>Completed: {item.completed ? 'Yes' : 'No'}</p>  </li>  ))}  </div>  </div>  );  }  export default App;  Output will be the same: Notice only that **Arrow function** with **curly braces** must include **return** |
|  |
|  |