**Explain how to consume REST APIs from React applications**

To consume REST APIs in a React application, developers typically use the native JavaScript fetch() API or popular libraries like axios. These tools allow the application to send HTTP requests (such as GET, POST, PUT, DELETE) to a server and receive responses. In React, API calls are often made inside the useEffect() hook so they can run when the component mounts. The response data is usually stored in a component’s state using the useState() hook.

For example, to fetch a list of users from an API, you can use fetch() to send a GET request and then update the state with the received data. The axios library simplifies syntax and handles some features like automatic JSON parsing, making it a preferred choice for many developers. To send data to an API, such as creating a new post, a POST request is made with appropriate headers and a JSON body.

Error handling is also important, so developers typically include .catch() blocks (for fetch) or try...catch when using async/await to handle any network errors. For more complex applications, custom hooks or state management libraries like React Query or Redux Toolkit Query can provide features like caching, automatic refetching, and background updates. Overall, consuming REST APIs in React involves integrating HTTP requests into the component lifecycle and managing the resulting data effectively within React’s state system.

Example: GET Request

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

function UserList() {

const [users, setUsers] = useState([]);

useEffect(() => {

fetch('https://jsonplaceholder.typicode.com/users')

.then(response => response.json())

.then(data => setUsers(data))

.catch(error => console.error('Error fetching users:', error));

}, []);

return (

<div>

<h2>User List</h2>

<ul>

{users.map(user => <li key={user.id}>{user.name}</li>)}

</ul>

</div>

);

}

export default UserList;

**Example: POST Request**

axios.post('https://jsonplaceholder.typicode.com/posts', {

title: 'New Post',

body: 'This post was created using axios.',

userId: 1

})

.then(response => console.log('Created:', response.data))

.catch(error => console.error('Error:', error));

**Create a React Application “fetchuserapp” which will retrieve the user details from** [**https://api.randomuser.me/**](https://api.randomuser.me/) **and display the title, firstname and image of a user.**

**Index.js :-**

import React from 'react';

import ReactDOM from 'react-dom/client';

import './index.css';

import App from './App';

import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(

  <React.StrictMode>

    <App />

  </React.StrictMode>

);

reportWebVitals();

**App.js :-**

import React from 'react';

import './App.css';

import Getuser from './Getuser';

function App() {

  return (

    <div className="App">

      <h1>Random User App</h1>

      <Getuser />

    </div>

  );

}

export default App;

**Getuser.js :-**

import React from 'react';

class Getuser extends React.Component {

  constructor(props) {

    super(props);

    this.state = {

      user: null

    };

  }

  async componentDidMount() {

    try {

      const response = await fetch('https://api.randomuser.me/');

      const data = await response.json();

      const user = data.results[0];

      this.setState({ user });

    } catch (error) {

      console.error('Error fetching user:', error);

    }

  }

  render() {

    const { user } = this.state;

    return (

      <div style={{ textAlign: 'center', marginTop: '50px' }}>

        {user ? (

          <div>

            <h2>

              {user.name.title} {user.name.first} {user.name.last}

            </h2>

            <img src={user.picture.medium} alt="User" />

          </div>

        ) : (

          <p>Loading user...</p>

        )}

      </div>

    );

  }

}

export default Getuser;

Output :-



