**Exercise 27: Demo about AXIOS**

Objectives and Outcomes

This exercise demonstrates a simple demo that showcases how to use Axios, a popular JavaScript library for making HTTP requests, in a React application.

Exercises

Create a new React component called AxiosDemo:

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

import axios from 'axios';

const AxiosDemo = () => {

const [data, setData] = useState(null);

const [error, setError] = useState(null);

useEffect(() => {

axios

.get('https://jsonplaceholder.typicode.com/posts/1')

.then(response => {

setData(response.data);

})

.catch(error => {

setError(error.message);

});

}, []);

if (error) {

return <div>Error: {error}</div>;

}

return (

<div>

<h1>Axios Demo</h1>

{data ? (

<div>

<h2>{data.title}</h2>

<p>{data.body}</p>

</div>

) : (

<div>Loading...</div>

)}

</div>

);

};

export default AxiosDemo;

In this component, we import Axios and the necessary React hooks (useState and useEffect). We initialize two state variables: data and error. The data state will store the fetched data, and the error state will hold any error messages.

Inside the useEffect hook, we make an HTTP GET request to the JSONPlaceholder API (<https://jsonplaceholder.typicode.com/posts/1>). If the request is successful, we set the data state with the response data. If there is an error, we set the error state with the error message.

In the JSX, we conditionally render the content based on the state. If there is an error, we display the error message. If data is available, we display the title and body of the fetched post. While waiting for the request to complete, we show a "Loading..." message.

To use the AxiosDemo component, render it in the root of your application:

import React from 'react';

import ReactDOM from 'react-dom';

import AxiosDemo from './AxiosDemo';

function App() {

return (

<React.StrictMode>

<AxiosDemo />

</React.StrictMode>

);

}

export default App;

Conclusion

This demo demonstrates how to use Axios in a React component to fetch data from an API. You can modify the URL to fetch different data or explore other Axios functionalities like POST, PUT, DELETE requests, request headers, and more.