DEPARTMENT OF CSE
COURSE CODE: 23SDCS12A / 23SDCS12R
FULL STACK APPLICATION DEVELOPMENT

Date of the Session: __/__/_____ **Time of The Session:_____to_____**

LAB – 6 ➔ Transferring (Sending and Receiving) data with Axios / Fetch API in React

Prerequisites:

Axios / Fetch need to be installed

Exercise 1:

Create a React component called UserData that fetches user data from a given API endpoint using Axios. Display the fetched user data in a table format.

Use this API end point to get user data - https://api.github.com/users/defunkt/following or https://reqres.in/api/users

Exercise 2:

Send API request to 3rd party and get the response to print it.

Use this API call - https://rapidapi.com.

For example weather api (use any api as your wish) -https://rapidapi.com/worldapi/api/open-weather13/playground/apiendpoint_d15cd885-e8e5-49e7-b94b-588c41687aa1

index.html

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <link rel="icon" type="image/svg+xml" href="/vite.svg" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Vite + React</title>
  </head>
  <body>
    <div id="root"></div>
    <script type="module" src="/src/main.jsx"></script>
  </body>
</html>
```

❖ Watch The Video And Do In Visual Studio

6a https://youtu.be/La6bWKK9bSE?si=FIVS6jFg16fyttNC

6b https://youtu.be/fmEoYjHG-gU?si=nXDJTtnpIDGO5gPe

Exercise 1

User.jsx

```
import axios from "axios";
import { useState, useEffect } from "react";

export default function User() {
  const [result, setResult] = useState("Nothing");

  useEffect(() => {
    if (result === "Nothing") {
      axios.get("https://reqres.in/api/users")
        .then((res) => {
          setResult(res.data.data);
          console.log(res.data.data);
        });
    }
  }, [result]);

  if (result === "Nothing") {
    return <div>Data is fetching</div>;
  }

  return (
    <div>
      This is User Page
      <br />
      <br />
      <table border={1}>
        <thead>
          <tr className="text-center" style={{ fontWeight: 'bold' }}>
            <td>ID</td>
            <td>EMAIL</td>
            <td>FIRST NAME</td>
            <td>LAST NAME</td>
            <td>IMAGE</td>
          </tr>
        </thead>
```

```
        <tbody>
         {result.map((element) => (
           <tr key={element.id}>
             <td>{element.id}</td>
             <td>{element.email}</td>
             <td>{element.first_name}</td>
             <td>{element.last_name}</td>
             <td><img src={element.avatar} alt="error" width={75} height={75} /></td>
           </tr>
         ))}
        </tbody>
       </table>
     </div>
   );
 }
```

main.jsx

```
import { StrictMode } from "react";
import { createRoot } from "react-dom/client";
import "./index.css";
import App from './App.jsx'
import User from "./components/lab 6/User.jsx";

createRoot(document.getElementById("root")).render(
 <StrictMode>
  <User/>
 </StrictMode>,
)
```

Output

This is User Page

ID	EMAIL	FIRST NAME	LAST NAME	IMAGE
1	george.bluth@reqres.in	George	Bluth	
2	janet.weaver@reqres.in	Janet	Weaver	
3	emma.wong@reqres.in	Emma	Wong	
4	eve.holt@reqres.in	Eve	Holt	
5	charles.morris@reqres.in	Charles	Morris	
6	tracey.ramos@reqres.in	Tracey	Ramos	

Exercise 2

Climate.jsx

```
import axios from "axios";
import { useState, useEffect } from "react";

export default function Climate() {
  const [result, setResult] = useState(null);

  useEffect(() => {
    if (result === null) {
      axios.get("https://open-weather13.p.rapidapi.com/city/vijayawada/EN", {
        headers: {
          'Accept': 'application/json',
          'Content-Type': 'application/json',
          'x-rapidapi-ua': 'RapidAPI-Playground',
          'x-rapidapi-key': '199baa6db6msh3b40c9b8e498a0bp1cde3bjsnc92bd606ad60',
          'x-rapidapi-host': 'open-weather13.p.rapidapi.com',
        }
      }).then((res) => {
        setResult(res.data.main.temp);
        console.log(res.data.main.temp);
      });
    }
  }, [result]);

  if (result === null) {
    return (
      <div>
        This is a Climate Display Page
        <br /><br />
        Data is fetching
      </div>
    );
  }

  return (
    <div>
```

```
      Vijayawada Temp is: {result}
    </div>
  );
 }
```

main.jsx

```
import { StrictMode } from "react";
import { createRoot } from "react-dom/client";
import "./index.css";
import App from './App.jsx'
import Climate from "./components/lab 6/Climate.jsx";

createRoot(document.getElementById("root")).render(
 <StrictMode>
  <Climate/>
 </StrictMode>,
)
```

Output

This is a Climate Display Page

Data is fetching

VIVA QUESTIONS:

1. How do you handle loading, success, and error states in a React component when making an API request with Axios or Fetch?

Loading, Success, and Error: Use `useState` for states, `try-catch` for errors, and update states accordingly.

2. Explain how you would use useEffect with Axios or Fetch to make API calls in a React functional component. What are the potential issues, and how would you prevent unnecessary re-renders?

`useEffect` **with Axios/Fetch**: Call API inside `useEffect`. Avoid re-renders by using dependency array `[]`.

3. How can you pass headers or authentication tokens with Axios or Fetch in a React application, and what's the best way to keep this information secure?

Passing Headers/Tokens: Use `headers` in Axios or Fetch requests. Store tokens securely in `httpOnly` cookies or `localStorage`.

4. How would you cancel an API request in React if the component unmounts before the request completes, using Axios or Fetch?

Cancel API Request: Use `axios.CancelToken` or `AbortController` to cancel requests on unmount.

5. How do you handle asynchronous operations with Axios / Fetch?

Async Operations: Use `async/await` for Axios/Fetch with `try-catch` for error handling.

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation Marks Secured _____ out of 50 Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation: