

## Lesson 04 Demo 03

### Create a React Application That Demonstrates useEffect Hook

**Objective:** To develop a React application that uses the useEffect Hook to fetch data

**Tools required:** Node terminal, React app, and Visual Studio Code

**Prerequisites:** Knowledge of creating a React app and understanding of the folder structure

Steps to be followed:

1. Create a new **React** app
2. Update the **App.js** file to use **useEffect** Hook
3. Run the app and view it in the browser

#### Step 1: Create a new React app

- 1.1 Open your terminal and run the **npx create-react-app useeffect-demo** command to create a React application with the name **useeffect-demo**

```
shreemayeebhatt@ip-172-31-22-250:~$ npx create-react-app useeffect-demo
```

```
Creating a new React app in /home/shreemayeebhatt/useeffect-demo.
```

- 1.2 Run the **cd useeffect-demo** command in the terminal to change the current directory to the newly created React app directory

```
Happy hacking!
```

```
shreemayeebhatt@ip-172-31-22-250:~$ cd useeffect-demo/
```

## Step 2: Update the App.js file to use the useEffect Hook

2.1 Open your React project in **Visual Studio Code**, navigate through the **useeffect-demo** project structure to open the **App.js** file within the **src** directory, and start modifying it

2.2 Import the **useState** and **useEffect** Hooks from the **React** library using the following code:

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

```
1 import logo from './logo.svg';  
2 import React, { useState, useEffect } from 'react';  
3 import './App.css';  
4
```

**Note:** This will make these Hooks available for use in the component.

2.3 Declare a state variable named **users** using the **useState** Hook and initialize it with an empty array

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

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

2.4 Use the **useEffect** Hook to fetch data from the specified API URL when the component mounts

```
useEffect(() => {  
  fetch('https://jsonplaceholder.typicode.com/users')  
    .then(response => response.json())  
    .then(data => setUsers(data))  
    .catch(error => console.log(error));  
}, []);
```

```
useEffect(() => {  
  fetch('https://jsonplaceholder.typicode.com/users')  
    .then(response => response.json())  
    .then(data => setUsers(data))  
    .catch(error => console.log(error));  
}, []);
```

**Note:**

1. Inside the **useEffect** Hook, the **fetch** function is used to send a **GET** request to the API URL and handle the response.
2. The response data is then converted into **JSON** format using the **.json()** method.
3. Update the **users** state with the received data using the **setUsers** function
4. Handle any potential errors by adding a **.catch()** method to the fetch request

2.5 In the return statement, display the data fetched from the API by using the following code:

```
return (  
  <div className="App">  
    <h1>useEffect Demo</h1>  
    <ul>  
      {users.map(user => (  
        <li key={user.id}>{user.name}</li>  
      ))}  
    </ul>  
  </div>  
}
```

export default App;

```

return (
  <div className="App">
    <h1>useEffect Demo</h1>
    <ul>
      {users.map(user => (
        <li key={user.id}>{user.name}</li>
      ))}
    </ul>
  </div>
);
}

export default App;

```

**Note:** Refer to the following code to configure the App.js file:

```

import React, { useState, useEffect } from 'react';
import './App.css';

function App() {
  const [users, setUsers] = useState([]);

  useEffect(() => {
    fetch('https://jsonplaceholder.typicode.com/users')
      .then(response => response.json())
      .then(data => setUsers(data))
      .catch(error => console.log(error));
  }, []);

  return (
    <div className="App">
      <h1>useEffect Demo</h1>
      <ul>
        {users.map(user => (
          <li key={user.id}>{user.name}</li>
        ))}
      </ul>
    </div>
  );
}

```

```
}
```

```
export default App;
```

```
import logo from './logo.svg';
import React, { useState, useEffect } from 'react';
import './App.css';

function App() {
  const [users, setUsers] = useState([]);

  useEffect(() => {
    fetch('https://jsonplaceholder.typicode.com/users')
      .then(response => response.json())
      .then(data => setUsers(data))
      .catch(error => console.log(error));
  }, []);

  return (
    <div className="App">
      <h1>useEffect Demo</h1>
      <ul>
        {users.map(user => (
          <li key={user.id}>{user.name}</li>
        ))}
      </ul>
    </div>
  );
}

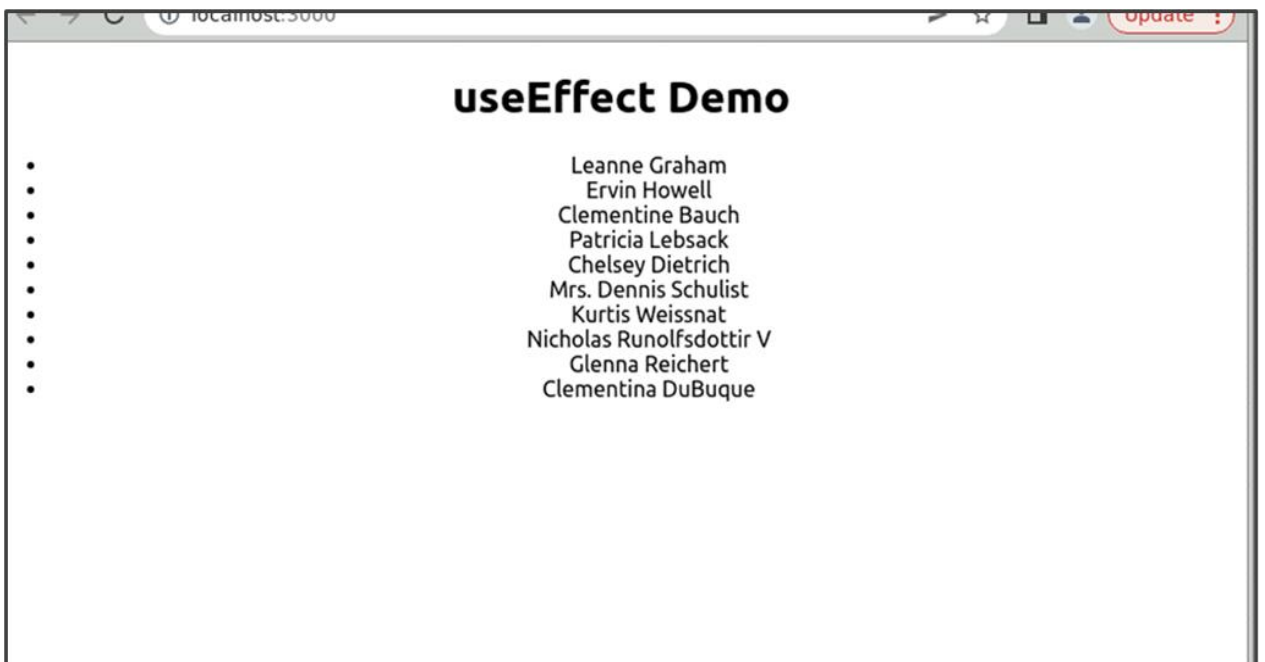
export default App;
```

### Step 3: Run the app and view it in the browser

- 3.1 In the terminal, navigate to the project directory and run the **npm start** command to start the app

```
shreemayeebhatt@ip-172-31-22-250:~/useeffect-demo$ npm start
```

- 3.2 Open your browser and navigate to **http://localhost:3000** to see the final output



You should see a simple app that displays a list of user names fetched from the **JSONPlaceholder API**.

With this, you've successfully created a React application that uses the **useEffect** Hook to fetch data.