React Assignment

1.

Home,js

```
import React from 'react';
import './Home.css';

const Home = ({ user }) => {
  return (
    <div className="container">
      <h2>Welcome, {user.username}!</h2>
      <p>Email: {user.email}</p>
      <p>Welcome to Home Page.</p>
    </div>
  );
};

export default Home;
```

Home.css

```
.container {
  max-width: 600px;
  margin: 0 auto;
  padding: 20px;
  border: 1px solid #ccc;
  border-radius: 5px;
  background-color: #f9f9f9;
}

h2 {
  text-align: center;
  margin-bottom: 20px;
}

p {
  margin-bottom: 10px;
}
```

Login.js

```
import React from 'react';
import './Login.css';
```

```jsx
const Login = ({ onLogin }) => {
  const handleLogin = () => {
    onLogin({ username: 'user123', email: 'user123@example.com' });
  };

  return (
    <div className="container">
      <h2>Login Page</h2>
      <button onClick={handleLogin}>Login</button>
    </div>
  );
};

export default Login;
```

Login.css

```css
.container {
  max-width: 400px;
  margin: 0 auto;
  padding: 20px;
  border: 1px solid #ccc;
  border-radius: 5px;
  background-color: #f9f9f9;
}

h2 {
  text-align: center;
  margin-bottom: 20px;
}

button {
  display: block;
  width: 100%;
  padding: 10px;
  background-color: #007bff;
  color: #fff;
  border: none;
  border-radius: 5px;
  cursor: pointer;
}

button:hover {
  background-color: #0056b3;
}
```

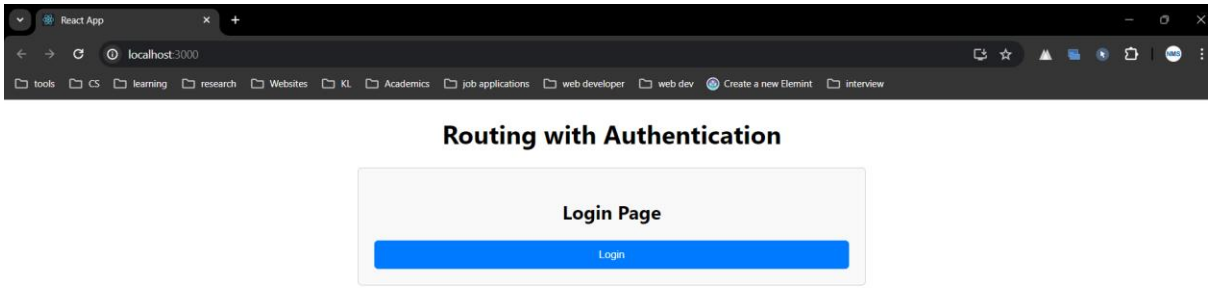
App.js

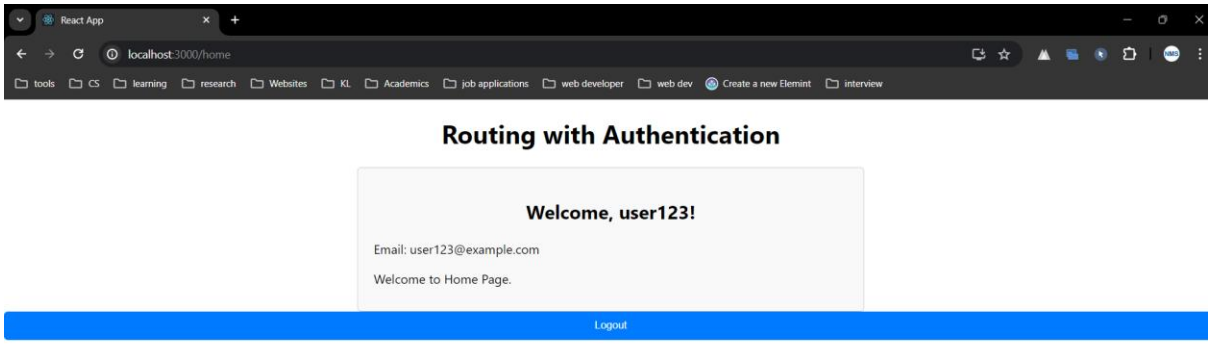
```
import React, { useState } from 'react';
import { BrowserRouter as Router, Routes, Route, Navigate } from 'react-
router-dom';
import Login from './components/Login';
import Home from './components/Home';

const App = () => {
  const [isLoggedIn, setIsLoggedIn] = useState(false);
  const [user, setUser] = useState(null);

  const handleLogin = (userData) => {
    setIsLoggedIn(true);
    setUser(userData);
  };

  const handleLogout = () => {
    setIsLoggedIn(false);
    setUser(null);
  };

  return (
    <Router>
      <div>
        <h1 style={{ textAlign: 'center' }}>Routing with Authentication</h1>
        <Routes>
          <Route
            path="/"
            element={isLoggedIn ? <Navigate to="/home" /> : <Login
onLogin={handleLogin} />}
          />
          <Route
            path="/home"
            element={isLoggedIn ? <Home user={user} /> : <Navigate to="/" />}
          />
        </Routes>
        {isLoggedIn && (
          <button onClick={handleLogout}>Logout</button>
        )}
      </div>
    </Router>
  );
};

export default App;
```

App.css

```css
.App {
  text-align: center;
}

.App-logo {
  height: 40vmin;
  pointer-events: none;
}

@media (prefers-reduced-motion: no-preference) {
  .App-logo {
    animation: App-logo-spin infinite 20s linear;
  }
}

.App-header {
  background-color: #282c34;
  min-height: 100vh;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  font-size: calc(10px + 2vmin);
  color: white;
}

.App-link {
  color: #61dafb;
}

@keyframes App-logo-spin {
  from {
    transform: rotate(0deg);
  }
  to {
    transform: rotate(360deg);
  }
}
```

Output:


React App
localhost:3000
tools
CS
learning
research
Websites
KL
Academics
job applications
web developer
web dev
Create a new Elemint
interview
Routing with Authentication
Login Page
Login


React App
localhost:3000/home
tools
CS
learning
research
Websites
KL
Academics
job applications
web developer
web dev
Create a new Elemint
interview
Routing with Authentication
Welcome, user123!
Email: user123@example.com
Welcome to Home Page.
Logout

2.

Dashboard.js

```
import React from "react";

const Dashboard = ({ username, users }) => {
  return (
    <div>
      <h2>Welcome, {username}!</h2>
      <h3>Logged-in Users:</h3>
      <ul>
        {users.map((user) => (
          <li key={user.username}>{user.username}</li>
        ))}
      </ul>
    </div>
  );
};

export default Dashboard;
```

Login.js:

```
import React, { useState } from "react";


const Login = ({ onLogin }) => {
  const [username, setUsername] = useState("");
  const [password, setPassword] = useState("");

  const handleLogin = (e) => {
    e.preventDefault();
    if (username === "admin" && password === "password") {
      onLogin(username);
    } else {
      alert("Invalid username or password");
    }
  };

  return (
    <div>
      <h2>Login</h2>
      <form onSubmit={handleLogin}>
        <div>
          <label>Username:</label>
          <input
            type="text"
```

```
            value={username}
            onChange={(e) => setUsername(e.target.value)}
            required
          />
        </div>
        <div>
          <label>Password:</label>
          <input
            type="password"
            value={password}
            onChange={(e) => setPassword(e.target.value)}
            required
          />
        </div>
        <button type="submit">Login</button>
      </form>
    </div>
  );
};

export default Login;
```

ParentComponent.js:

```
import React, { useState, useEffect } from "react";
import Dashboard from "./Dashboard";

const ParentComponent = () => {
  const [users, setUsers] = useState([]);

  useEffect(() => {
    const fetchUsers = async () => {
      try {
        const response = await fetch("https://api.example.com/users");
        const data = await response.json();
        setUsers(data);
      } catch (error) {
        console.error("Error fetching users:", error);
      }
    };

    fetchUsers();
  }, []);

  return (
    <div>
      <h1>User Management System</h1>
```

```
      <Dashboard users={users} />
    </div>
  );
};

export default ParentComponent;
```

Signup.js:

```
import React, { useState } from "react";


const Signup = ({ onSignup }) => {
  const [username, setUsername] = useState("");
  const [password, setPassword] = useState("");

  const handleSignup = (e) => {
    e.preventDefault();
    const newUser = { username, password };
    onSignup(newUser);
    setUsername("");
    setPassword("");
  };

  return (
    <div>
      <h2>Signup</h2>
      <form onSubmit={handleSignup}>
        <div>
          <label>Username:</label>
          <input
            type="text"
            value={username}
            onChange={(e) => setUsername(e.target.value)}
            required
          />
        </div>
        <div>
          <label>Password:</label>
          <input
            type="password"
            value={password}
            onChange={(e) => setPassword(e.target.value)}
            required
          />
        </div>
        <button type="submit">Signup</button>
```

```
      </form>
    </div>
  );
};

export default Signup;
```

App.js:

```
import React, { useState } from "react";
import {
  BrowserRouter as Router,
  Routes,
  Route,
  Navigate,
} from "react-router-dom";
import Login from "./components/Login";
import Signup from "./components/Signup";
import Dashboard from "./components/Dashboard";

const App = () => {
  const [loggedInUser, setLoggedInUser] = useState(null);
  const [users, setUsers] = useState([]);

  const handleLogin = (username) => {
    setLoggedInUser(username);
    setUsers([...users, { username }]);
  };

  const handleSignup = (newUser) => {
    console.log("New user registered:", newUser);
  };

  return (
    <Router>
      <div>
        <h1>Login and Signup App</h1>
        <Routes>
          <Route
            path="/"
            element={
              loggedInUser ? (
                <Navigate to="/dashboard" />
              ) : (
                <Login onLogin={handleLogin} />
              )
            }
          />
```

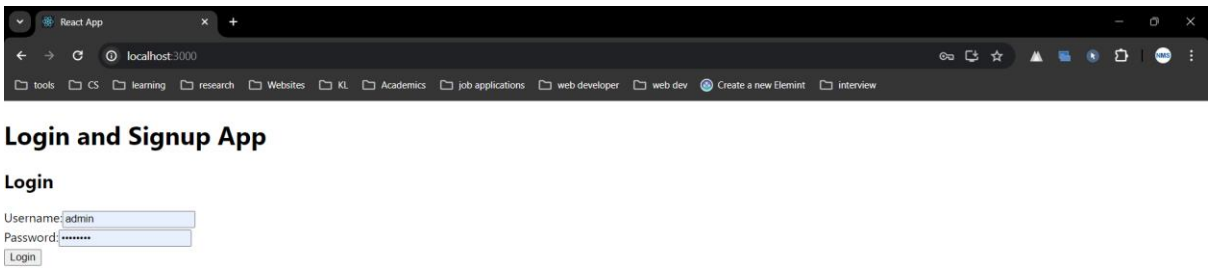
```
          <Route path="/signup" element={<Signup onSignup={handleSignup} />}
/>
          <Route
            path="/dashboard"
            element={
              loggedInUser ? (
                <Dashboard username={loggedInUser} users={users} />
              ) : (
                <Navigate to="/" />
              )
            }
          />
        </Routes>
      </div>
    </Router>
  );
};

export default App;
```

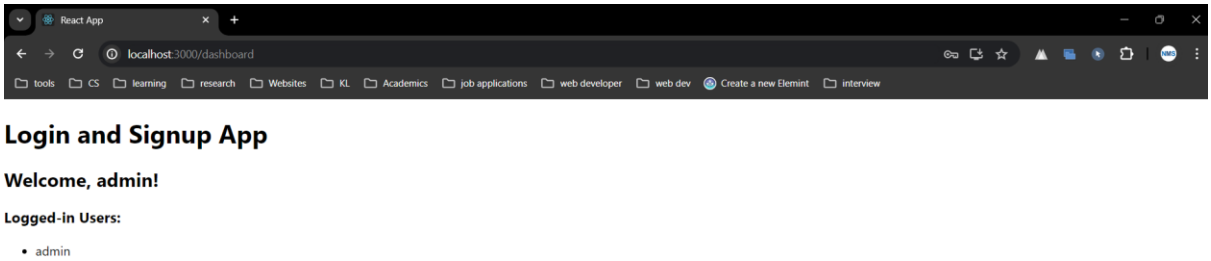
App.css:

```
.App {
  text-align: center;
}

.App-logo {
  height: 40vmin;
  pointer-events: none;
}

@media (prefers-reduced-motion: no-preference) {
  .App-logo {
    animation: App-logo-spin infinite 20s linear;
  }
}

.App-header {
  background-color: #282c34;
  min-height: 100vh;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  font-size: calc(10px + 2vmin);
  color: white;
```

```
}

.App-link {
  color: #61dafb;
}

@keyframes App-logo-spin {
  from {
    transform: rotate(0deg);
  }
  to {
    transform: rotate(360deg);
  }
}
```

output:





3.

FunctionalLifecycleComponent.js

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

const FunctionalLifecycleComponent = () => {
  const [count, setCount] = useState(0);

  useEffect(() => {
    console.log("Component did mount (functional component)");
    return () => {
      console.log("Component will unmount (functional component)");
    };
  }, []);

  useEffect(() => {
    console.log("Count state updated (functional component)");
  }, [count]);

  const incrementCount = () => {
    setCount((prevCount) => prevCount + 1);
  };

  console.log("Render executed (functional component)");

  return (
    <div>
      <h2>Functional Component with useEffect Hook</h2>
```

```
      <p>Count: {count}</p>
      <button onClick={incrementCount}>Increment Count</button>
    </div>
  );
};

export default FunctionalLifecycleComponent;
```

LifecycleComponent.js:

```
import React, { Component } from "react";

class LifecycleComponent extends Component {
  constructor(props) {
    super(props);
    this.state = {
      count: 0,
    };
    console.log("Constructor executed");
  }

  componentDidMount() {
    console.log("Component did mount");
  }

  componentDidUpdate(prevProps, prevState) {
    console.log("Component did update");
  }

  componentWillUnmount() {
    console.log("Component will unmount");
  }

  incrementCount = () => {
    this.setState((prevState) => ({
      count: prevState.count + 1,
    }));
  };

  render() {
    console.log("Render executed");
    return (
      <div>
        <h2>Component with Lifecycle Methods</h2>
        <p>Count: {this.state.count}</p>
        <button onClick={this.incrementCount}>Increment Count</button>
```

```
      </div>
    );
  }
}

export default LifecycleComponent;
```

App.js:

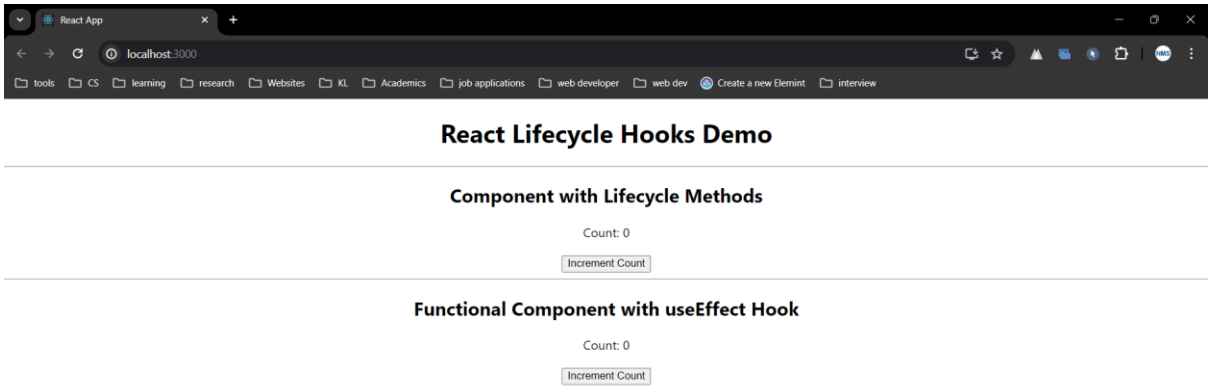
```
import React from "react";
import "./App.css";
import LifecycleComponent from "./LifecycleComponent";
import FunctionalLifecycleComponent from "./FunctionalLifecycleComponent";

function App() {
  return (
    <div className="App">
      <h1>React Lifecycle Hooks Demo</h1>
      <hr />
      <LifecycleComponent />
      <hr />
      <FunctionalLifecycleComponent />
    </div>
  );
}

export default App;
```

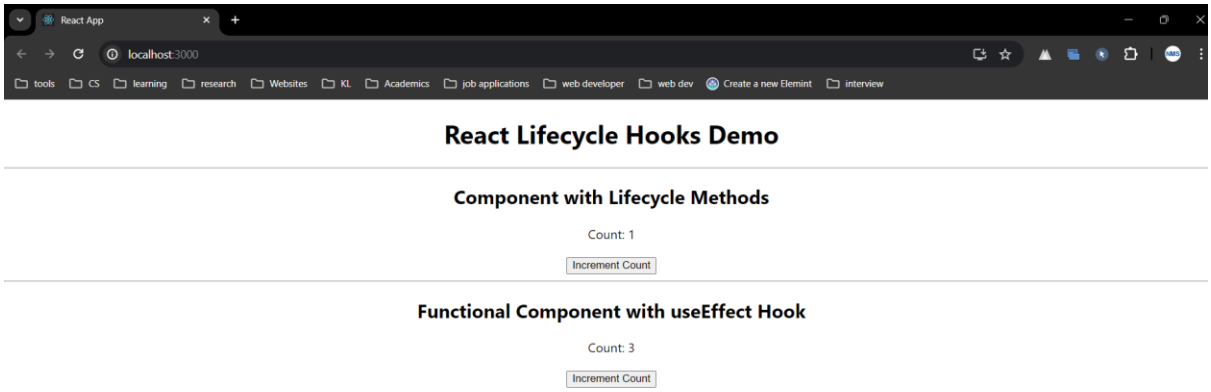
App.css:

```
.App {
  text-align: center;
}

.App-logo {
  height: 40vmin;
  pointer-events: none;
}

@media (prefers-reduced-motion: no-preference) {
  .App-logo {
    animation: App-logo-spin infinite 20s linear;
  }
```

```
}

.App-header {
  background-color: #282c34;
  min-height: 100vh;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  font-size: calc(10px + 2vmin);
  color: white;
}

.App-link {
  color: #61dafb;
}

@keyframes App-logo-spin {
  from {
    transform: rotate(0deg);
  }
  to {
    transform: rotate(360deg);
  }
}
```

Output:





4.

ProductItem.js

```
import React from "react";

const ProductItem = ({ product }) => {
  return (
    <div>
      <h3>{product.name}</h3>
      <p>Price: ${product.price}</p>
    </div>
  );
};

export default ProductItem;
```

ProductList.js

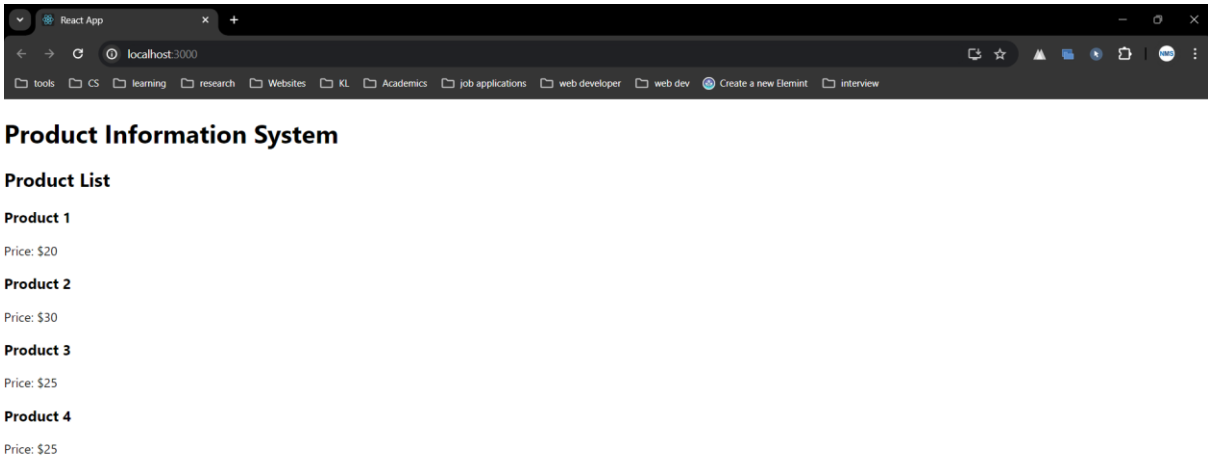
```
import React from "react";
import ProductItem from "./ProductItem";

const ProductList = () => {
  // Sample product data
```

```
  const products = [
    { id: 1, name: "Product 1", price: 20 },
    { id: 2, name: "Product 2", price: 30 },
    { id: 3, name: "Product 3", price: 25 },
    { id: 4, name: "Product 4", price: 25 },
  ];

  return (
    <div>
      <h2>Product List</h2>
      {products.map((product) => (
        <ProductItem key={product.id} product={product} />
      ))}
    </div>
  );
};

export default ProductList;
```

App.js:

```
import React from "react";
import ProductList from "./ProductList";

const App = () => {
  return (
    <div>
      <h1>Product Information System</h1>
      <ProductList />
    </div>
  );
};

export default App;
```

Output:



5.

PostWithAxios.js:

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

const PostWithAxios = () => {
  const [formData, setFormData] = useState({
    title: "",
    body: "",
  });

  const handleInputChange = (e) => {
    const { name, value } = e.target;
    setFormData({ ...formData, [name]: value });
  };

  const handleSubmit = async (e) => {
    e.preventDefault();

    try {
      const response = await axios.post(
        "https://jsonplaceholder.typicode.com/posts",
        formData
      );
      console.log("POST request successful:", response.data);
    } catch (error) {
```

```
      console.error("Error submitting post:", error);
    }
  };

  return (
    <div>
      <h2>Submit a Post using Axios</h2>
      <form onSubmit={handleSubmit}>
        <div>
          <label>Title:</label>
          <input
            type="text"
            name="title"
            value={formData.title}
            onChange={handleInputChange}
            required
          />
        </div>
        <div>
          <label>Body:</label>
          <textarea
            name="body"
            value={formData.body}
            onChange={handleInputChange}
            required
          />
        </div>
        <button type="submit">Submit</button>
      </form>
    </div>
  );
};

export default PostWithAxios;
```

PostWithFetch.js:

```
import React, { useState } from "react";

const PostWithFetch = () => {
  const [formData, setFormData] = useState({
    title: "",
    body: "",
  });

  const handleInputChange = (e) => {
```

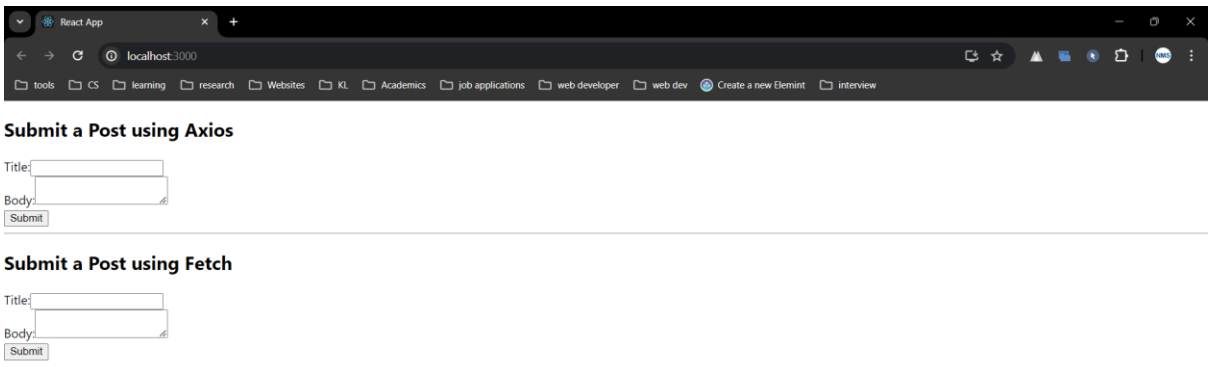
```
  const { name, value } = e.target;
  setFormData({ ...formData, [name]: value });
};

const handleSubmit = async (e) => {
  e.preventDefault();

  try {
    const response = await fetch(
      "https://jsonplaceholder.typicode.com/posts",
      {
        method: "POST",
        headers: {
          "Content-Type": "application/json",
        },
        body: JSON.stringify(formData),
      }
    );

    const data = await response.json();
    console.log("POST request successful:", data);
  } catch (error) {
    console.error("Error submitting post:", error);
  }
};

return (
  <div>
    <h2>Submit a Post using Fetch</h2>
    <form onSubmit={handleSubmit}>
      <div>
        <label>Title:</label>
        <input
          type="text"
          name="title"
          value={formData.title}
          onChange={handleInputChange}
          required
        />
      </div>
      <div>
        <label>Body:</label>
        <textarea
          name="body"
          value={formData.body}
          onChange={handleInputChange}
          required
        />
```

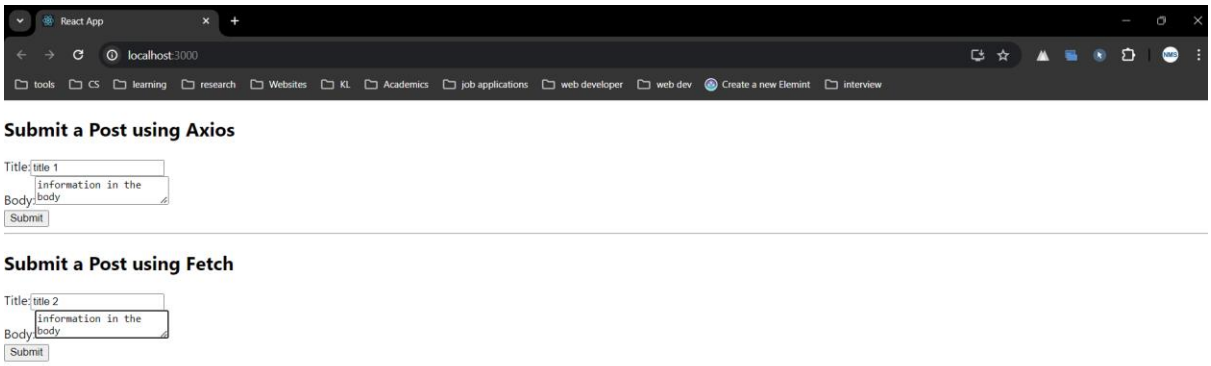
```
        </div>
        <button type="submit">Submit</button>
      </form>
    </div>
  );
};

export default PostWithFetch;
```

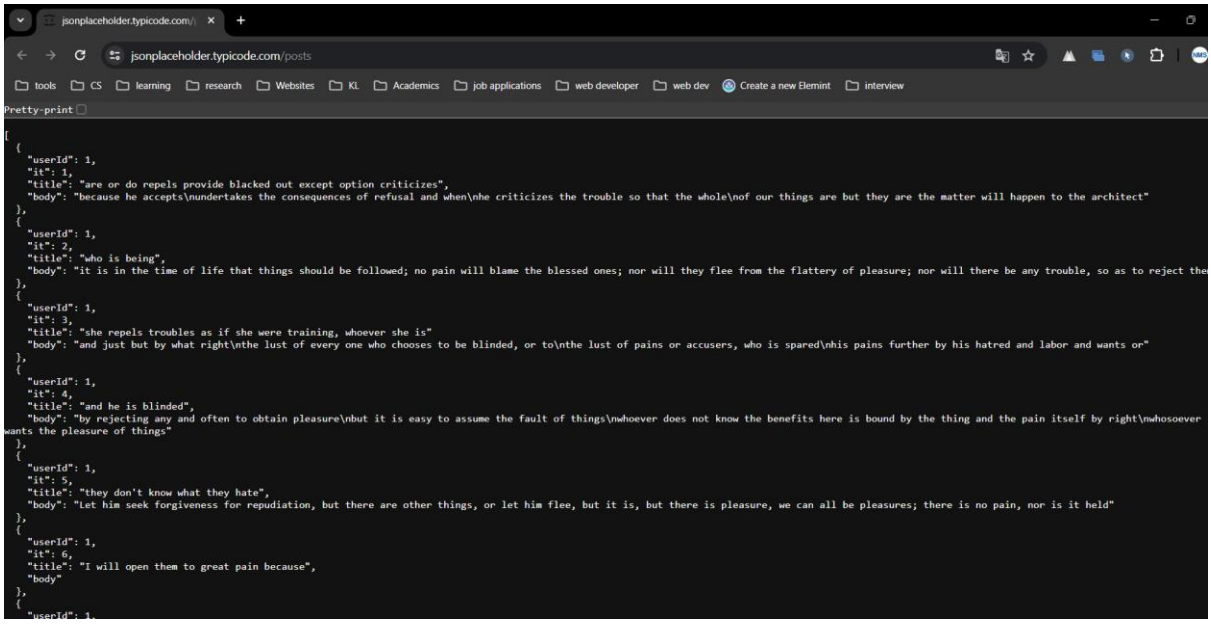
App.js:

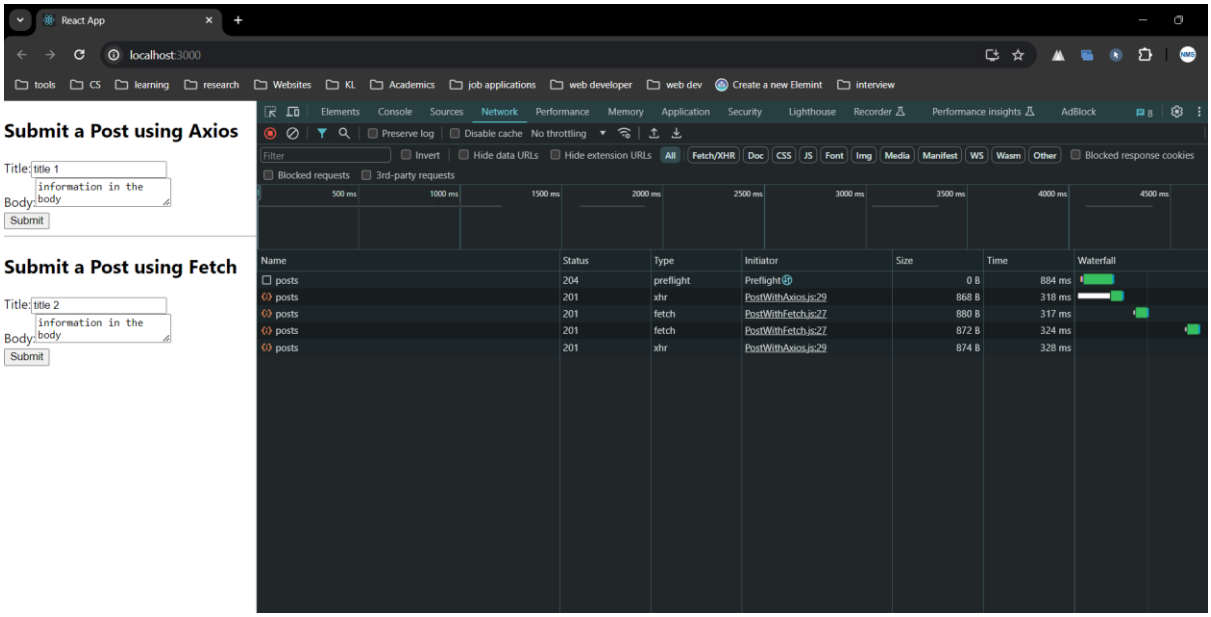
```
import React from "react";
import "./App.css";
import PostWithAxios from "./PostWithAxios";
import PostWithFetch from "./PostWithFetch";

function App() {
  return (
    <div className="App">
      <PostWithAxios />
      <hr />
      <PostWithFetch />
    </div>
  );
}

export default App;
```

output:




React App
localhost:3000
tools
CS
learning
research
Websites
KL
Academics
job applications
web developer
web dev
Create a new Elemint
interview
Submit a Post using Axios
Title:
title 1
Body:
information in the body
Submit
Submit a Post using Fetch
Title:
title 2
Body:
information in the body
Submit


jsonplaceholder.typicode.com/posts
tools
CS
learning
research
Websites
KL
Academics
job applications
web developer
web dev
Create a new Elemint
interview
Pretty-print
[
{
"userId": 1,
"it": 1,
"title": "are or do repels provide blacked out except option criticizes",
"body": "because he accepts\nundertakes the consequences of refusal and when\nhe criticizes the trouble so that the whole\nof our things are but they are the matter will happen to the architect"
},
{
"userId": 1,
"it": 2,
"title": "who is being",
"body": "it is in the time of life that things should be followed; no pain will blame the blessed ones; nor will they flee from the flattery of pleasure; nor will there be any trouble, so as to reject the
},
{
"userId": 1,
"it": 3,
"title": "she repels troubles as if she were training, whoever she is"
"body": "and just but by what right\nthe lust of every one who chooses to be blinded, or to\nthe lust of pains or accusers, who is spared\nhis pains further by his hatred and labor and wants or"
},
{
"userId": 1,
"it": 4,
"title": "and he is blinded",
"body": "by rejecting any and often to obtain pleasure\nbut it is easy to assume the fault of things\nwhoever does not know the benefits here is bound by the thing and the pain itself by right\nwhosoever
wants the pleasure of things"
},
{
"userId": 1,
"it": 5,
"title": "they don't know what they hate",
"body": "Let him seek forgiveness for repudiation, but there are other things, or let him flee, but it is, but there is pleasure, we can all be pleasures; there is no pain, nor is it held"
},
{
"userId": 1,
"it": 6,
"title": "I will open them to great pain because",
"body"
},
{


React App
localhost:3000
tools
CS
learning
research
Websites
KL
Academics
job applications
web developer
web dev
Create a new Elemint
interview
Submit a Post using Axios
Title:
title 1
Body:
information in the body
Submit
Submit a Post using Fetch
Title:
title 2
Body:
information in the body
Submit
Elements
Console
Sources
Network
Performance
Memory
Application
Security
Lighthouse
Recorder
Performance insights
AdBlock
Preserve log
Disable cache
No throttling
Filter
Invert
Hide data URLs
Hide extension URLs
All
Fetch/XHR
Doc
CSS
JS
Font
Img
Media
Manifest
WS
Wasm
Other
Blocked response cookies
Blocked requests
3rd-party requests
500 ms
1000 ms
1500 ms
2000 ms
2500 ms
3000 ms
3500 ms
4000 ms
4500 ms
Name
Status
Type
Initiator
Size
Time
Waterfall
posts 204 preflight Preflight 0 B 884 ms
posts 201 xhr PostWithAxios.js:29 868 B 318 ms
posts 201 fetch PostWithFetch.js:27 880 B 317 ms
posts 201 fetch PostWithFetch.js:27 872 B 324 ms
posts 201 xhr PostWithAxios.js:29 874 B 328 ms