Name: Ariba Shuaib

Registration Number: 230201075

Department and batch: CS O4 (A)

Semester: 5th semester

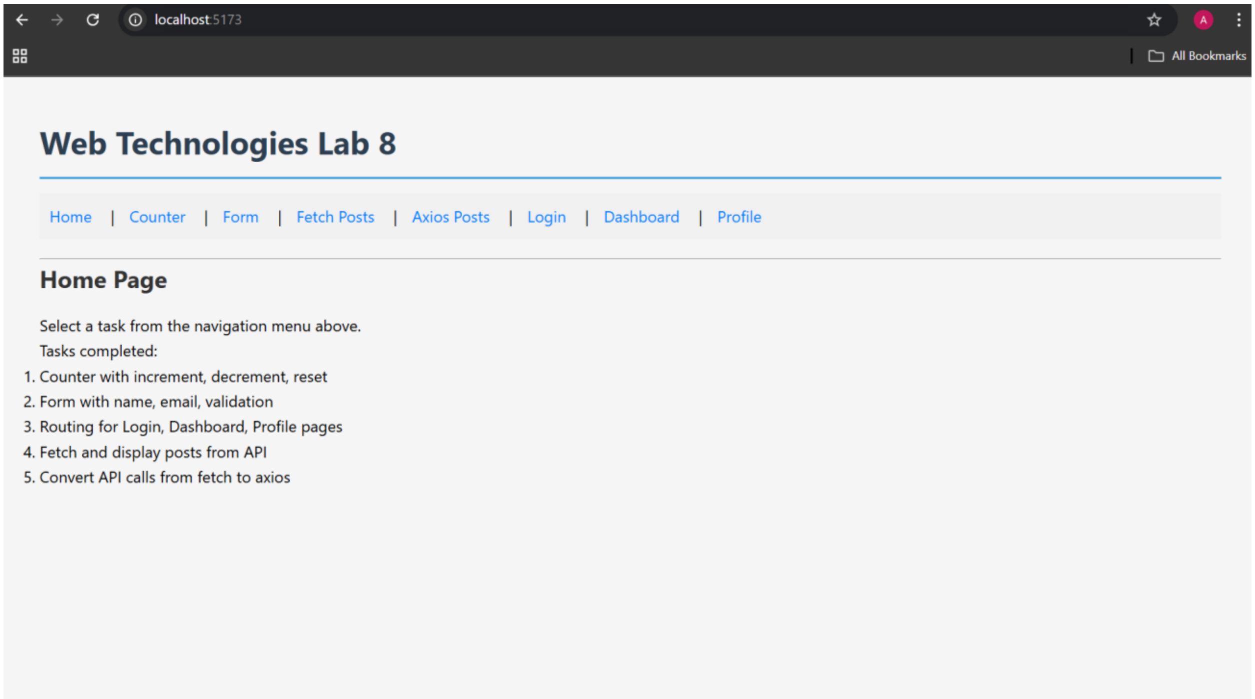
LAB REPORT 8

WEB TECHNOLOGIES

Institute of Space and Technology

Submitted to: Miss Maryam

Home:


localhost:5173
All Bookmarks
Home | Counter | Form | Fetch Posts | Axios Posts | Login | Dashboard | Profile

Web Technologies Lab 8

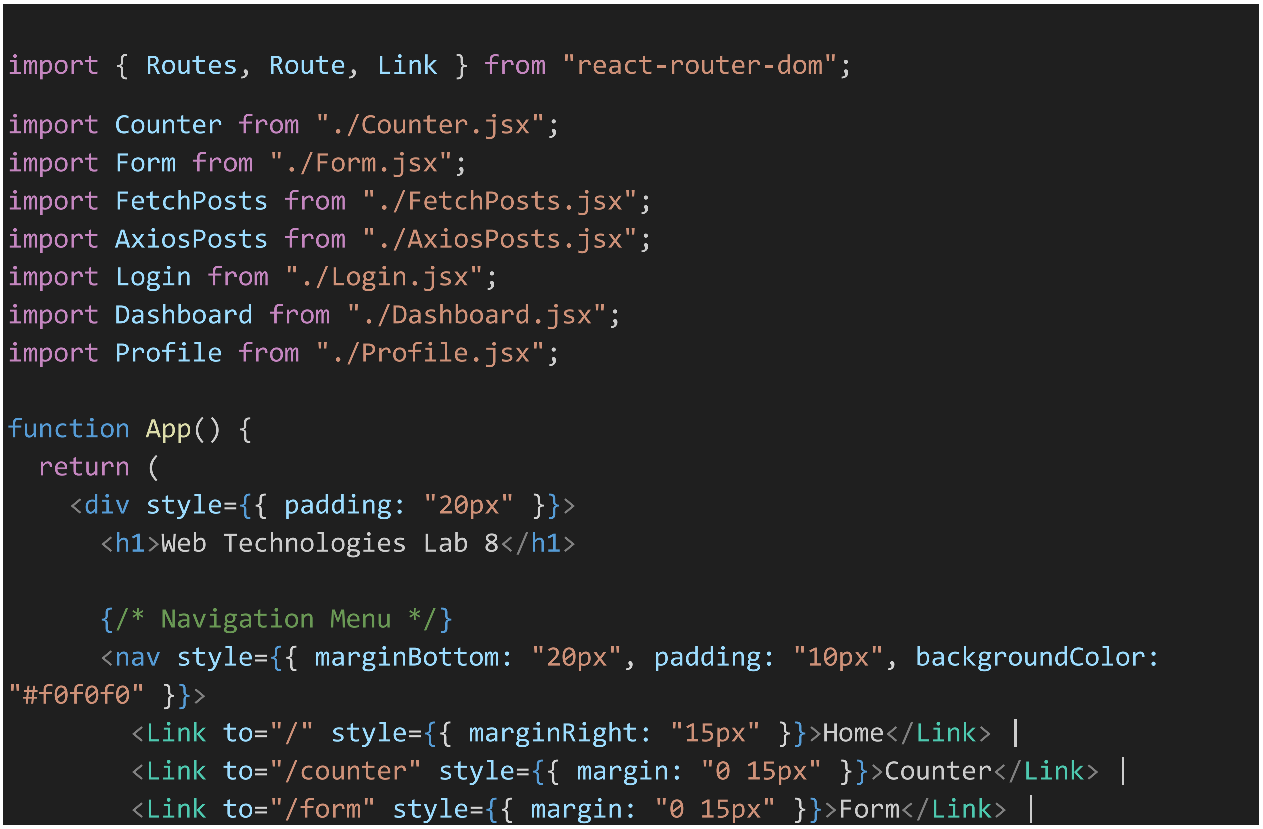
Home Page

Select a task from the navigation menu above.

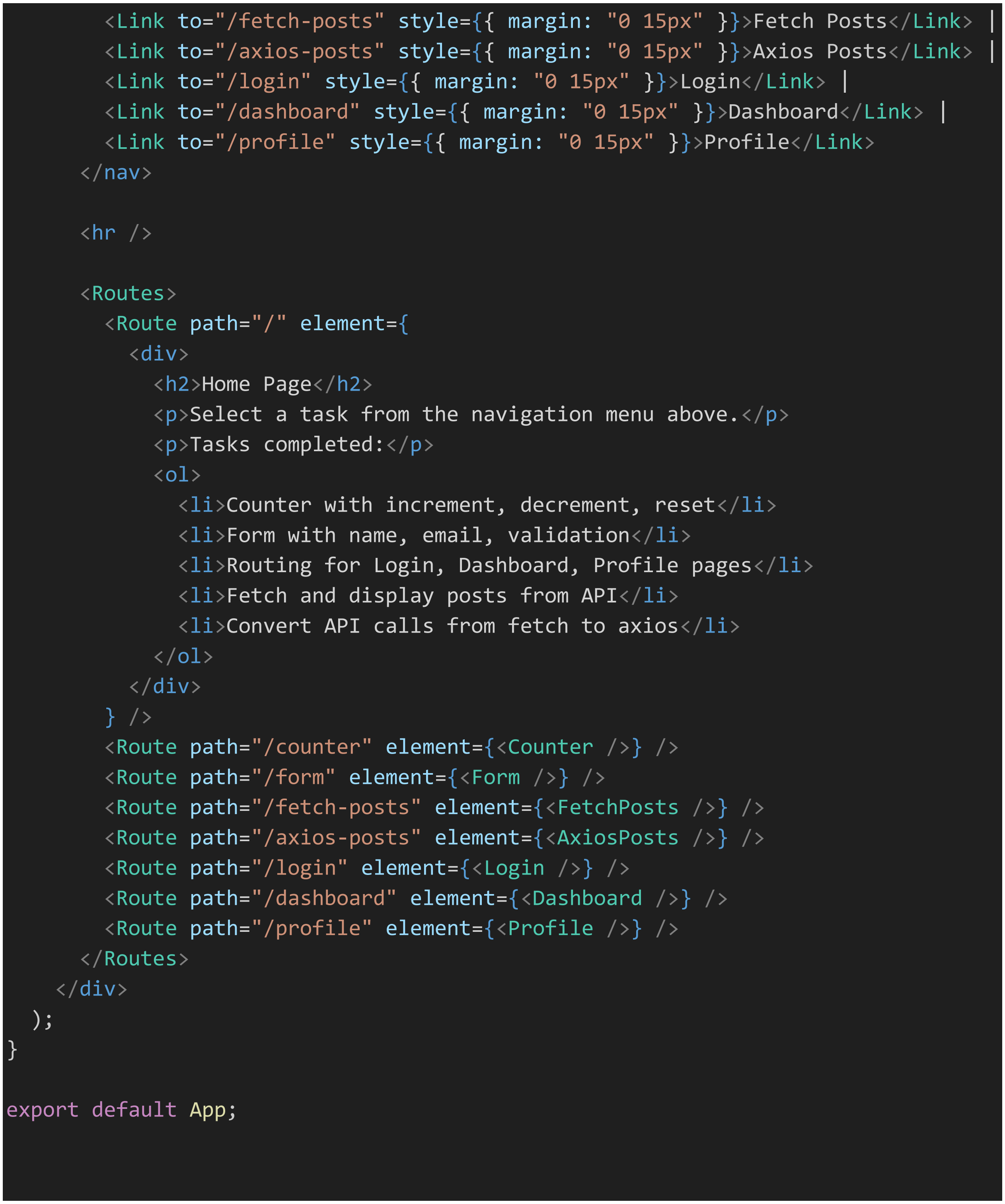
Tasks completed:

1. Counter with increment, decrement, reset
2. Form with name, email, validation
3. Routing for Login, Dashboard, Profile pages
4. Fetch and display posts from API
5. Convert API calls from fetch to axios

App.jsx:



```
import { Routes, Route, Link } from "react-router-dom";

import Counter from "./Counter.jsx";
import Form from "./Form.jsx";
import FetchPosts from "./FetchPosts.jsx";
import AxiosPosts from "./AxiosPosts.jsx";
import Login from "./Login.jsx";
import Dashboard from "./Dashboard.jsx";
import Profile from "./Profile.jsx";

function App() {
  return (
    <div style={{ padding: "20px" }}>
      <h1>Web Technologies Lab 8</h1>

      {/* Navigation Menu */}
      <nav style={{ marginBottom: "20px", padding: "10px", backgroundColor:
"#f0f0f0" }}>
        <Link to="/" style={{ marginRight: "15px" }}>Home</Link> |
        <Link to="/counter" style={{ margin: "0 15px" }}>Counter</Link> |
        <Link to="/form" style={{ margin: "0 15px" }}>Form</Link> |
```

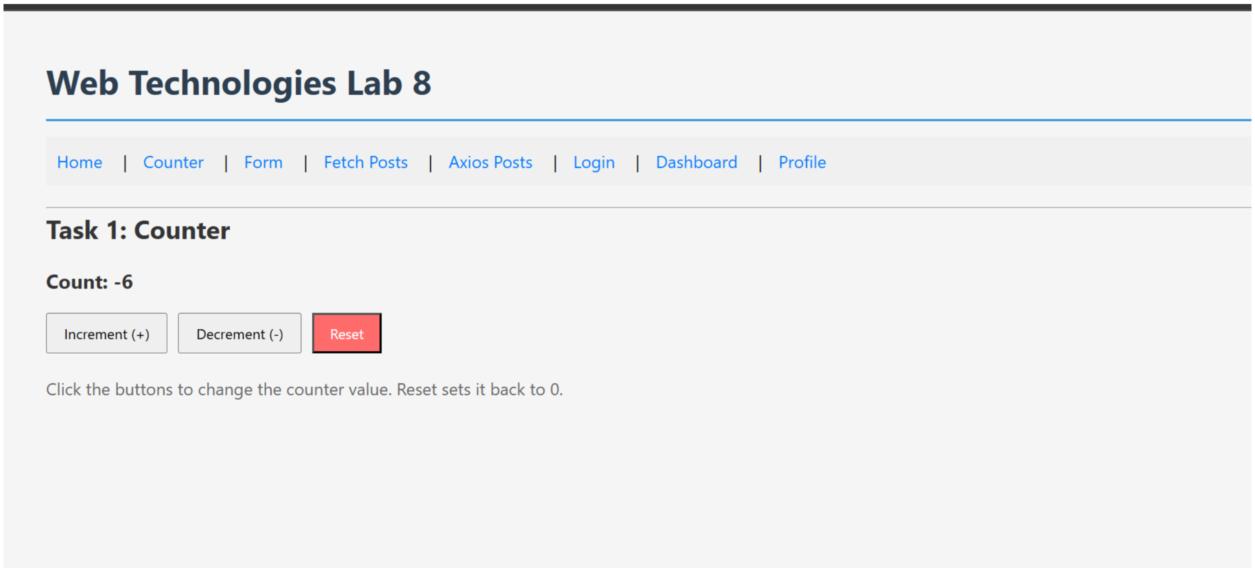


```
        <Link to="/fetch-posts" style={{ margin: "0 15px" }}>Fetch Posts</Link> |
        <Link to="/axios-posts" style={{ margin: "0 15px" }}>Axios Posts</Link> |
        <Link to="/login" style={{ margin: "0 15px" }}>Login</Link> |
        <Link to="/dashboard" style={{ margin: "0 15px" }}>Dashboard</Link> |
        <Link to="/profile" style={{ margin: "0 15px" }}>Profile</Link>
      </nav>

      <hr />

      <Routes>
        <Route path="/" element={
          <div>
            <h2>Home Page</h2>
            <p>Select a task from the navigation menu above.</p>
            <p>Tasks completed:</p>
            <ol>
              <li>Counter with increment, decrement, reset</li>
              <li>Form with name, email, validation</li>
              <li>Routing for Login, Dashboard, Profile pages</li>
              <li>Fetch and display posts from API</li>
              <li>Convert API calls from fetch to axios</li>
            </ol>
          </div>
        } />
        <Route path="/counter" element={<Counter />} />
        <Route path="/form" element={<Form />} />
        <Route path="/fetch-posts" element={<FetchPosts />} />
        <Route path="/axios-posts" element={<AxiosPosts />} />
        <Route path="/login" element={<Login />} />
        <Route path="/dashboard" element={<Dashboard />} />
        <Route path="/profile" element={<Profile />} />
      </Routes>
    </div>
  );
}

export default App;
```

Task1: Create a counter with increment, decrement, and reset

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

function Counter() {
  const [count, setCount] = useState(0);

  const increment = () => setCount(count + 1);
  const decrement = () => setCount(count - 1);
  const reset = () => setCount(0);

  return (
    <div>
      <h2>Task 1: Counter</h2>
      <h3>Count: {count}</h3>
      <button onClick={increment} style={{ marginRight: "10px", padding: "8px 15px" }}>
        Increment (+)
      </button>
      <button onClick={decrement} style={{ marginRight: "10px", padding: "8px 15px" }}>
        Decrement (-)
      </button>
      <button onClick={reset} style={{ padding: "8px 15px", backgroundColor: "#ff6b6b", color: "white" }}>
        Reset
      </button>
      <p style={{ marginTop: "20px", color: "#666" }}>
        Click the buttons to change the counter value. Reset sets it back to 0.
      </p>
    </div>
  );
}

export default Counter;
```

Output:



Web Technologies Lab 8

Home | Counter | Form | Fetch Posts | Axios Posts | Login | Dashboard | Profile

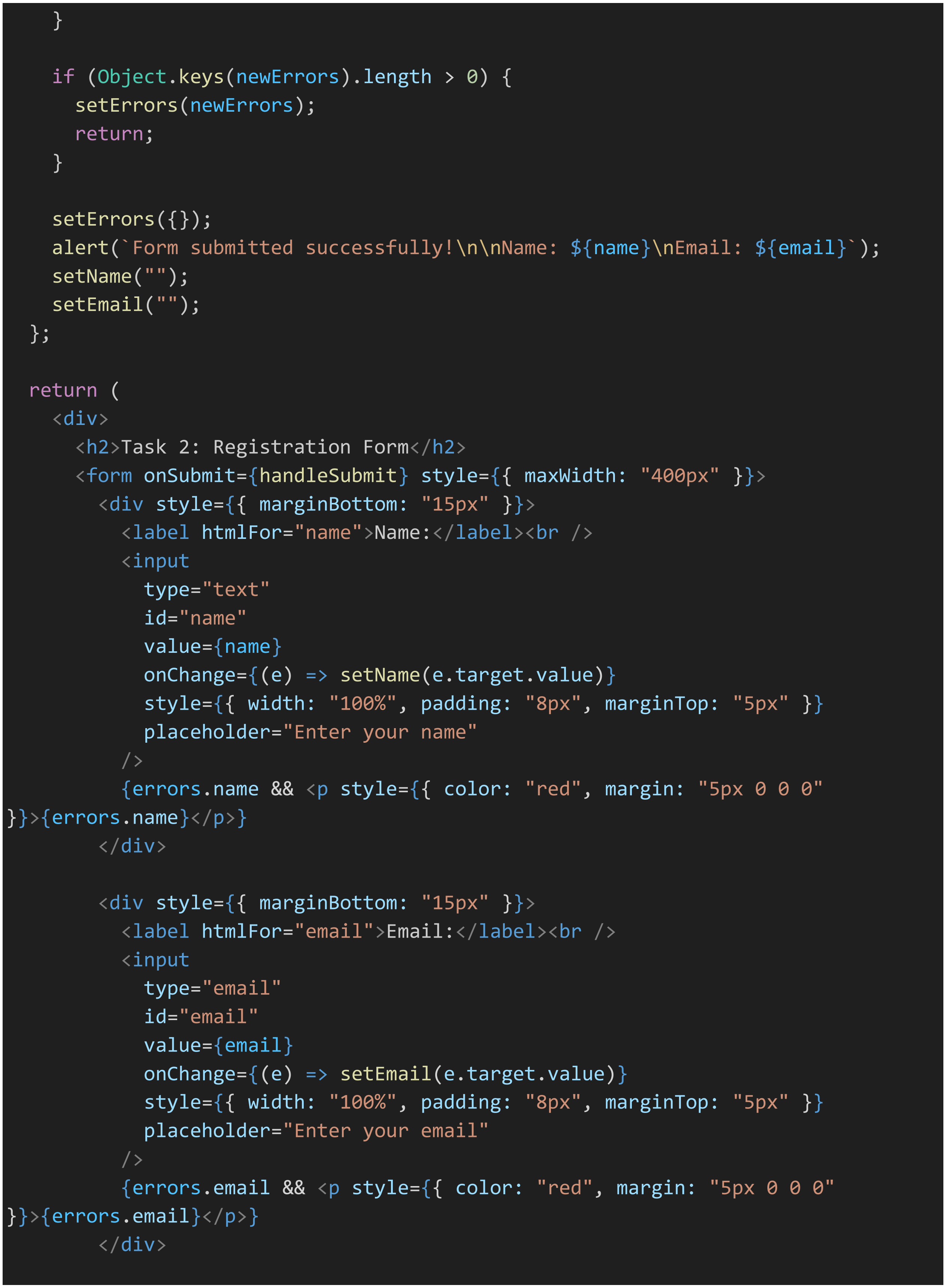
Task 1: Counter

Count: -6

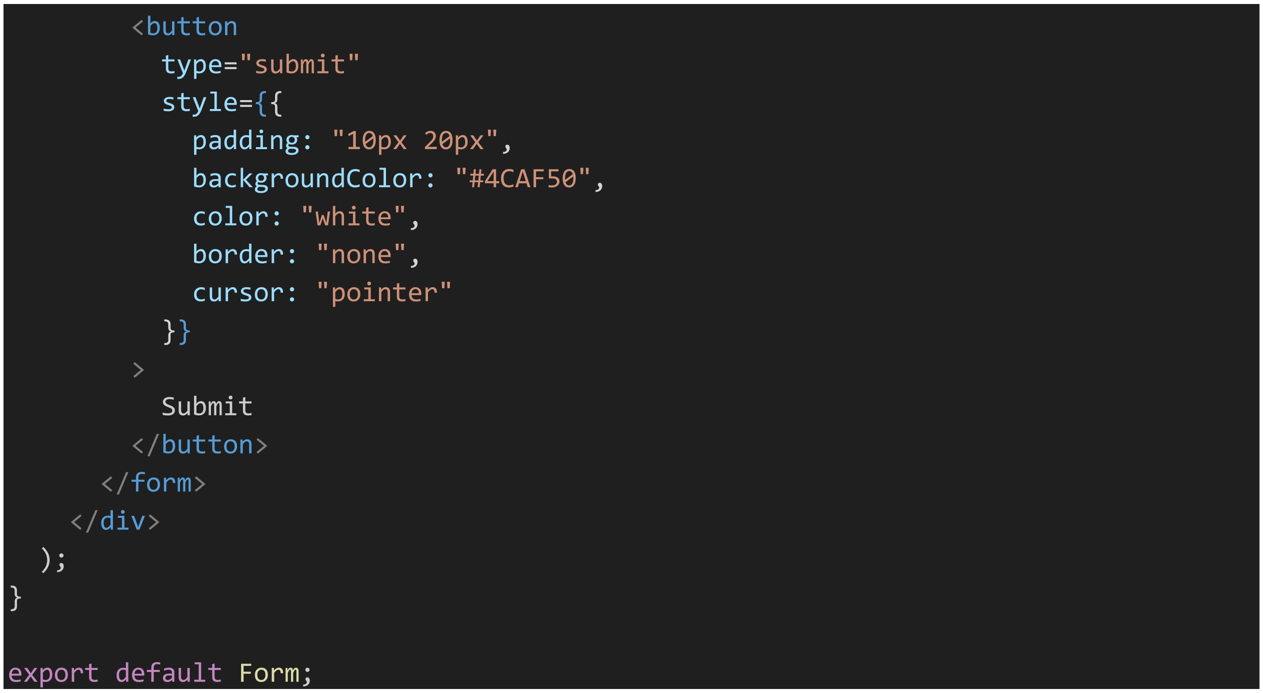
Click the buttons to change the counter value. Reset sets it back to 0.

Task 2: Build a form with name, email, and validation

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

function Form() {
  const [name, setName] = useState("");
  const [email, setEmail] = useState("");
  const [errors, setErrors] = useState({});

  const validateEmail = (email) => {
    const regex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
    return regex.test(email);
  };

  const handleSubmit = (e) => {
    e.preventDefault();
    const newErrors = {};

    if (!name.trim()) {
      newErrors.name = "Name is required";
    }

    if (!email.trim()) {
      newErrors.email = "Email is required";
    } else if (!validateEmail(email)) {
      newErrors.email = "Please enter a valid email";
```

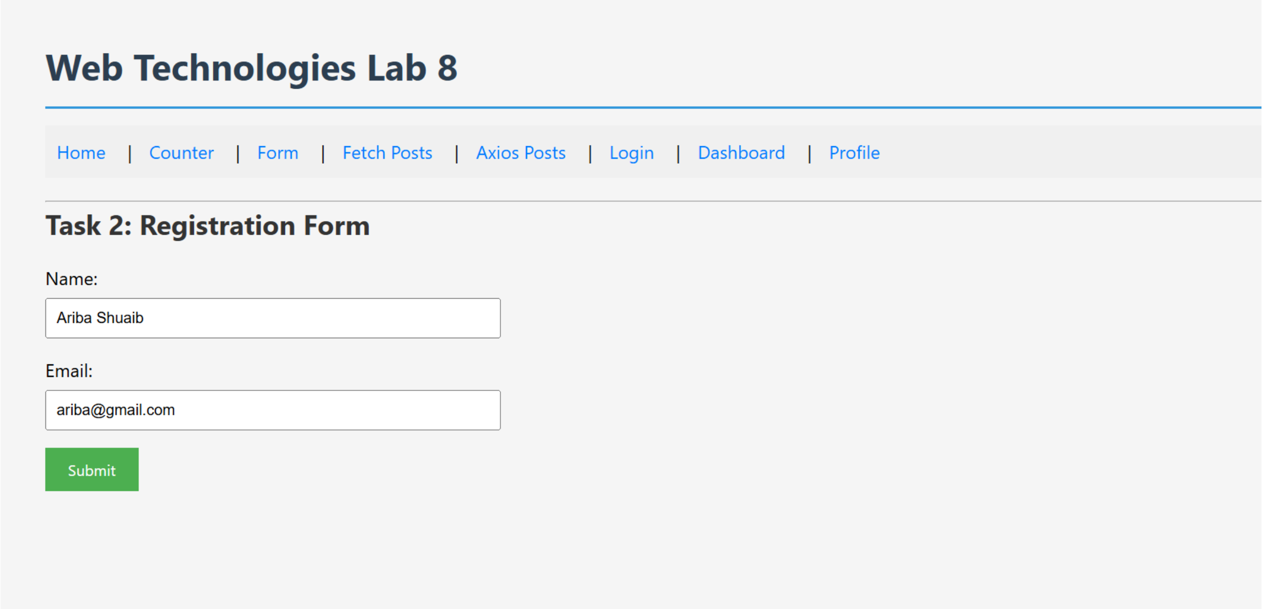


```
    }

    if (Object.keys(newErrors).length > 0) {
      setErrors(newErrors);
      return;
    }

    setErrors({});
    alert(`Form submitted successfully!\n\nName: ${name}\nEmail: ${email}`);
    setName("");
    setEmail("");
  };

  return (
    <div>
      <h2>Task 2: Registration Form</h2>
      <form onSubmit={handleSubmit} style={{ maxWidth: "400px" }}>
        <div style={{ marginBottom: "15px" }}>
          <label htmlFor="name">Name:</label><br />
          <input
            type="text"
            id="name"
            value={name}
            onChange={(e) => setName(e.target.value)}
            style={{ width: "100%", padding: "8px", marginTop: "5px" }}
            placeholder="Enter your name"
          />
          {errors.name && <p style={{ color: "red", margin: "5px 0 0 0"
}}>{errors.name}</p>}
        </div>

        <div style={{ marginBottom: "15px" }}>
          <label htmlFor="email">Email:</label><br />
          <input
            type="email"
            id="email"
            value={email}
            onChange={(e) => setEmail(e.target.value)}
            style={{ width: "100%", padding: "8px", marginTop: "5px" }}
            placeholder="Enter your email"
          />
          {errors.email && <p style={{ color: "red", margin: "5px 0 0 0"
}}>{errors.email}</p>}
        </div>
```



```
        <button
          type="submit"
          style={{
            padding: "10px 20px",
            backgroundColor: "#4CAF50",
            color: "white",
            border: "none",
            cursor: "pointer"
          }}
        >
          Submit
        </button>
      </form>
    </div>
  );
}

export default Form;
```

Output:



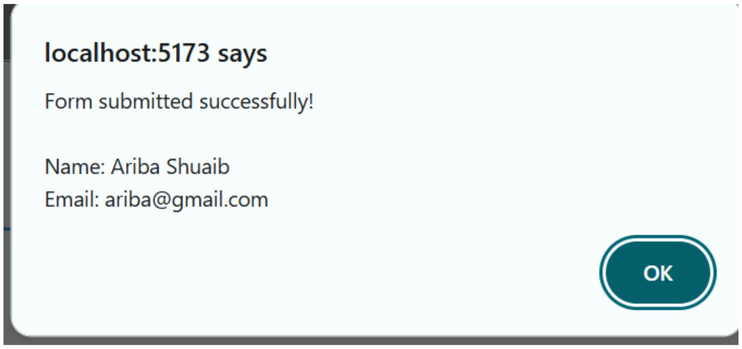
Web Technologies Lab 8

Home | Counter | Form | Fetch Posts | Axios Posts | Login | Dashboard | Profile

Task 2: Registration Form

Name: Ariba Shuaib

Email: ariba@gmail.com



localhost:5173 says

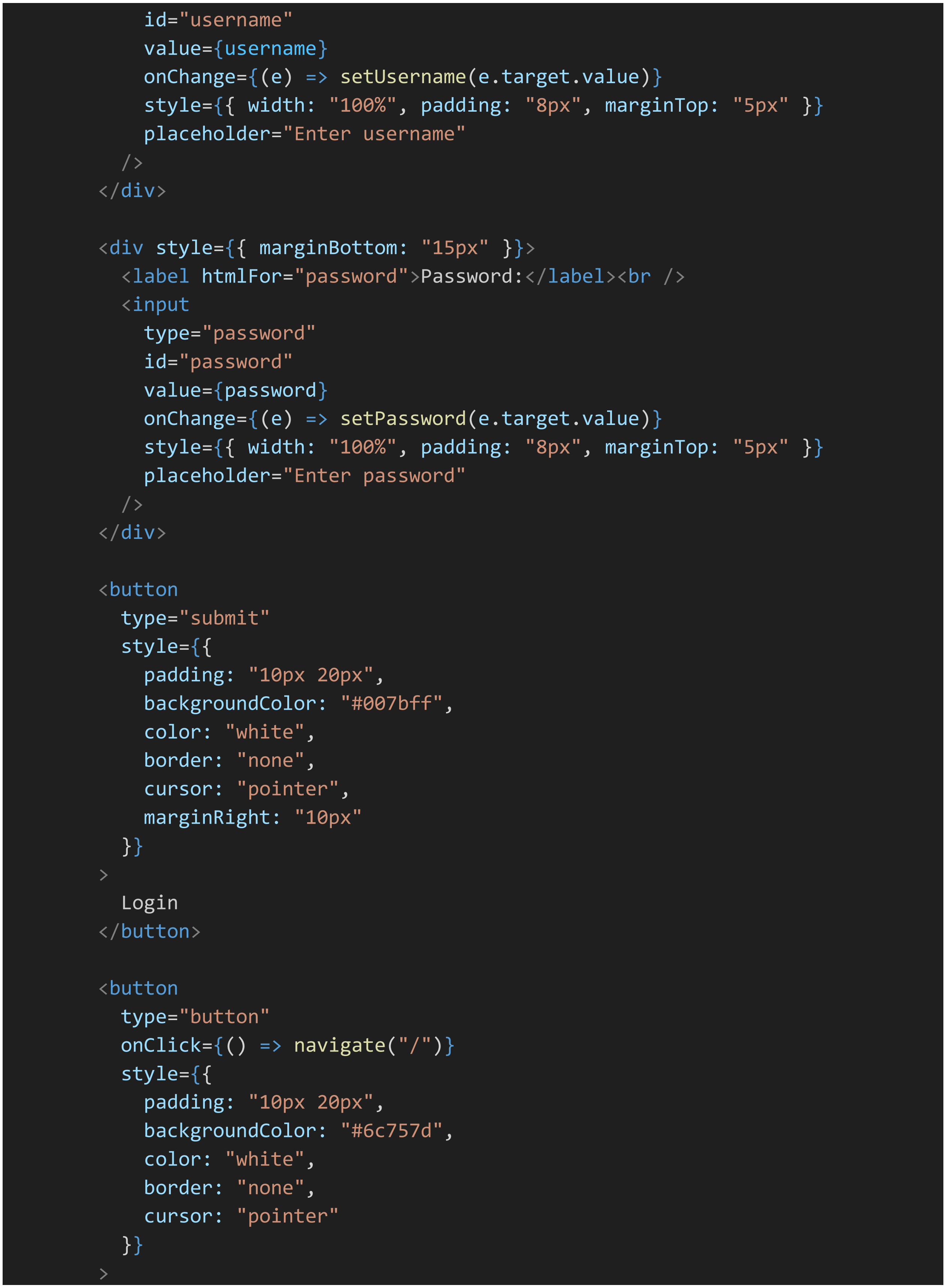
Form submitted successfully!

Name: Ariba Shuaib
Email: ariba@gmail.com

Task 3 — Implement routing for Login, Dashboard, and Profile pages

Login

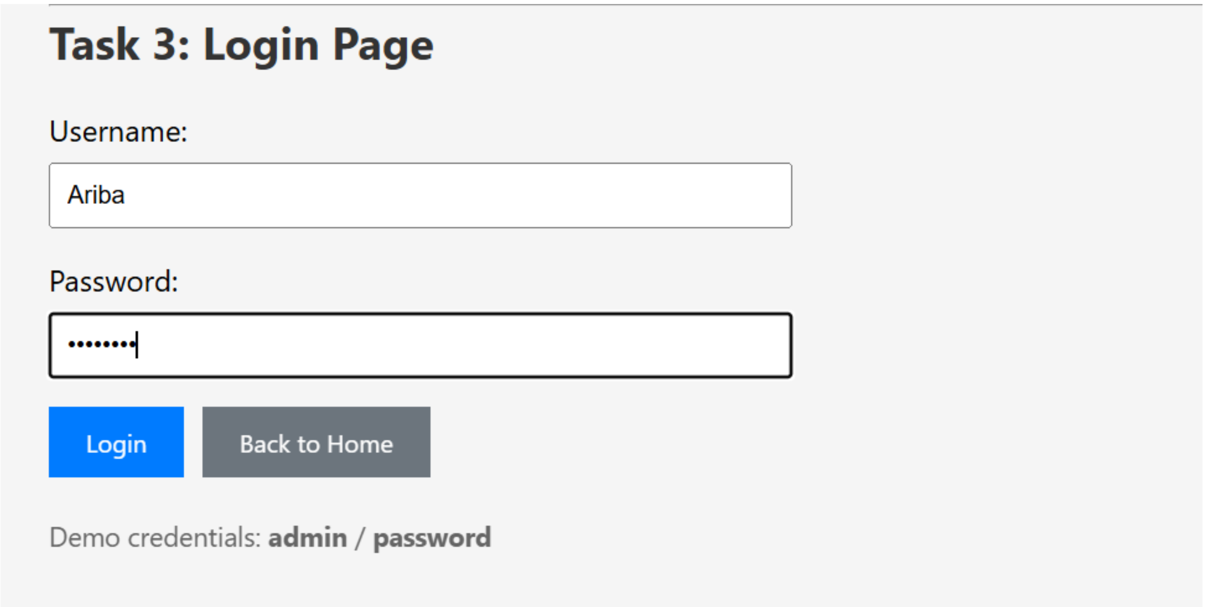
```
import { useState } from "react";
import { useNavigate } from "react-router-dom";

function Login() {
  const [username, setUsername] = useState("");
  const [password, setPassword] = useState("");
  const [error, setError] = useState("");
  const navigate = useNavigate();

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

    // Simple validation
    if (!username || !password) {
      setError("Both fields are required");
      return;
    }

    // Mock login - in real app, you'd call an API
    if (username === "admin" && password === "password") {
      setError("");
      alert("Login successful! Redirecting to dashboard...");
      navigate("/dashboard");
    } else {
      setError("Invalid credentials. Try admin/password");
    }
  };

  return (
    <div>
      <h2>Task 3: Login Page</h2>
      <form onSubmit={handleSubmit} style={{ maxWidth: "400px" }}>
        {error && <p style={{ color: "red", backgroundColor: "#ffe6e6", padding: "10px" }}>{error}</p>}

        <div style={{ marginBottom: "15px" }}>
          <label htmlFor="username">Username:</label><br />
          <input
            type="text"
```



```
            id="username"
            value={username}
            onChange={(e) => setUsername(e.target.value)}
            style={{ width: "100%", padding: "8px", marginTop: "5px" }}
            placeholder="Enter username"
          />
        </div>

        <div style={{ marginBottom: "15px" }}>
          <label htmlFor="password">Password:</label><br />
          <input
            type="password"
            id="password"
            value={password}
            onChange={(e) => setPassword(e.target.value)}
            style={{ width: "100%", padding: "8px", marginTop: "5px" }}
            placeholder="Enter password"
          />
        </div>

        <button
          type="submit"
          style={{
            padding: "10px 20px",
            backgroundColor: "#007bff",
            color: "white",
            border: "none",
            cursor: "pointer",
            marginRight: "10px"
          }}
        >
          Login
        </button>

        <button
          type="button"
          onClick={() => navigate("/")}
          style={{
            padding: "10px 20px",
            backgroundColor: "#6c757d",
            color: "white",
            border: "none",
            cursor: "pointer"
          }}
        >
```

```
          Back to Home
        </button>

        <p style={{ marginTop: "20px", color: "#666", fontSize: "0.9em" }}>
          Demo credentials: <strong>admin</strong> / <strong>password</strong>
        </p>
      </form>
    </div>
  );
}


export default Login;
```

Output:



Task 3: Login Page

Username:

Ariba

Password:

••••••••

Login

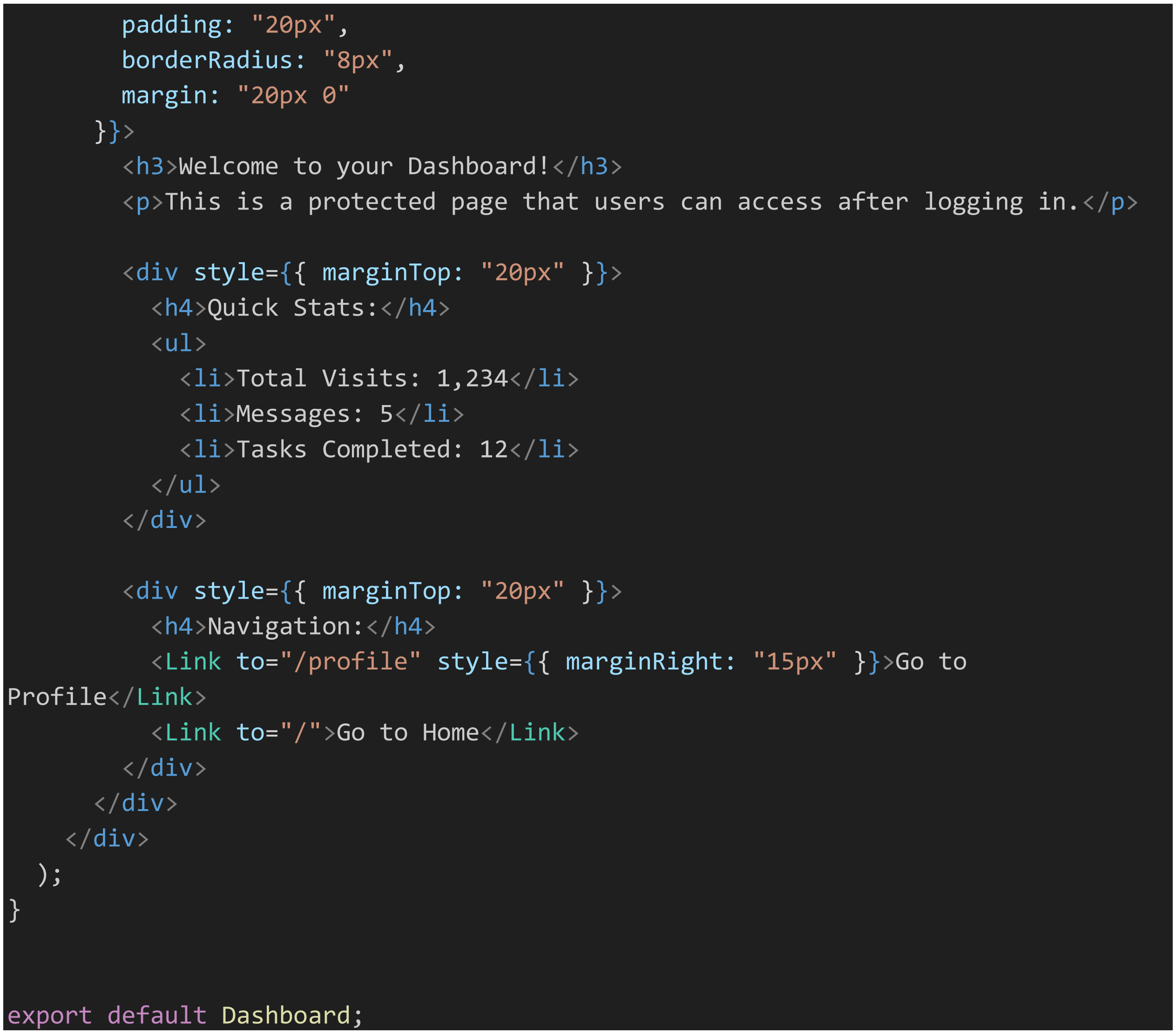
Back to Home

Demo credentials: **admin** / **password**

Dashboard

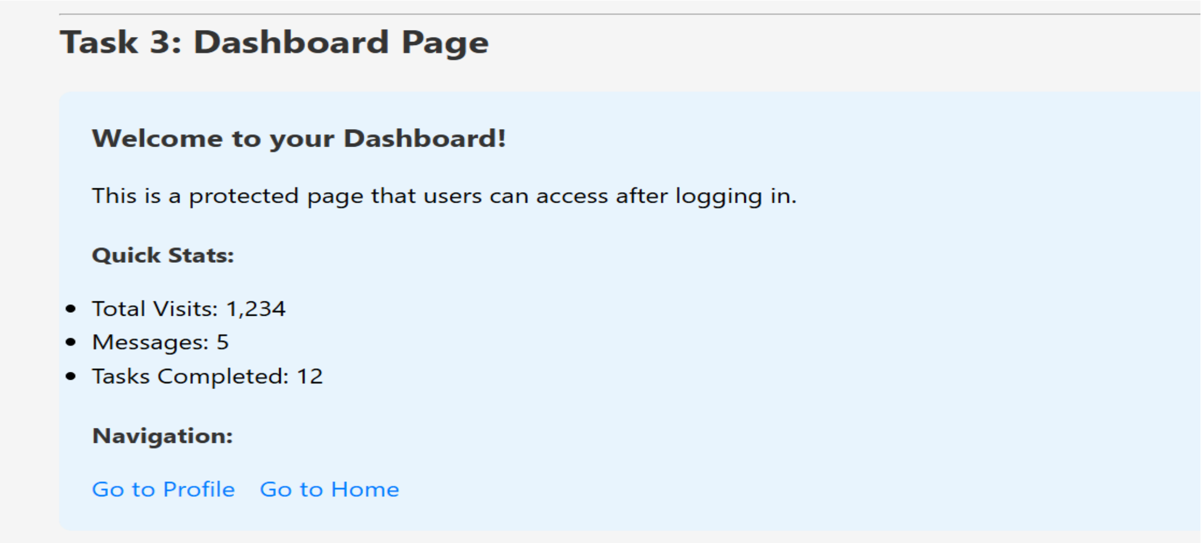
```
import { Link } from "react-router-dom";
```

```
function Dashboard() {
  return (
    <div>
      <h2>Task 3: Dashboard Page</h2>
      <div style={{
        backgroundColor: "#e8f4fd",
```



```
        padding: "20px",
        borderRadius: "8px",
        margin: "20px 0"
      }}>
        <h3>Welcome to your Dashboard!</h3>
        <p>This is a protected page that users can access after logging in.</p>

        <div style={{ marginTop: "20px" }}>
          <h4>Quick Stats:</h4>
          <ul>
            <li>Total Visits: 1,234</li>
            <li>Messages: 5</li>
            <li>Tasks Completed: 12</li>
          </ul>
        </div>

        <div style={{ marginTop: "20px" }}>
          <h4>Navigation:</h4>
          <Link to="/profile" style={{ marginRight: "15px" }}>Go to
Profile</Link>
          <Link to="/">Go to Home</Link>
        </div>
      </div>
    </div>
  );
}


export default Dashboard;
```

<u>Output:</u>



Task 3: Dashboard Page

Welcome to your Dashboard!

This is a protected page that users can access after logging in.

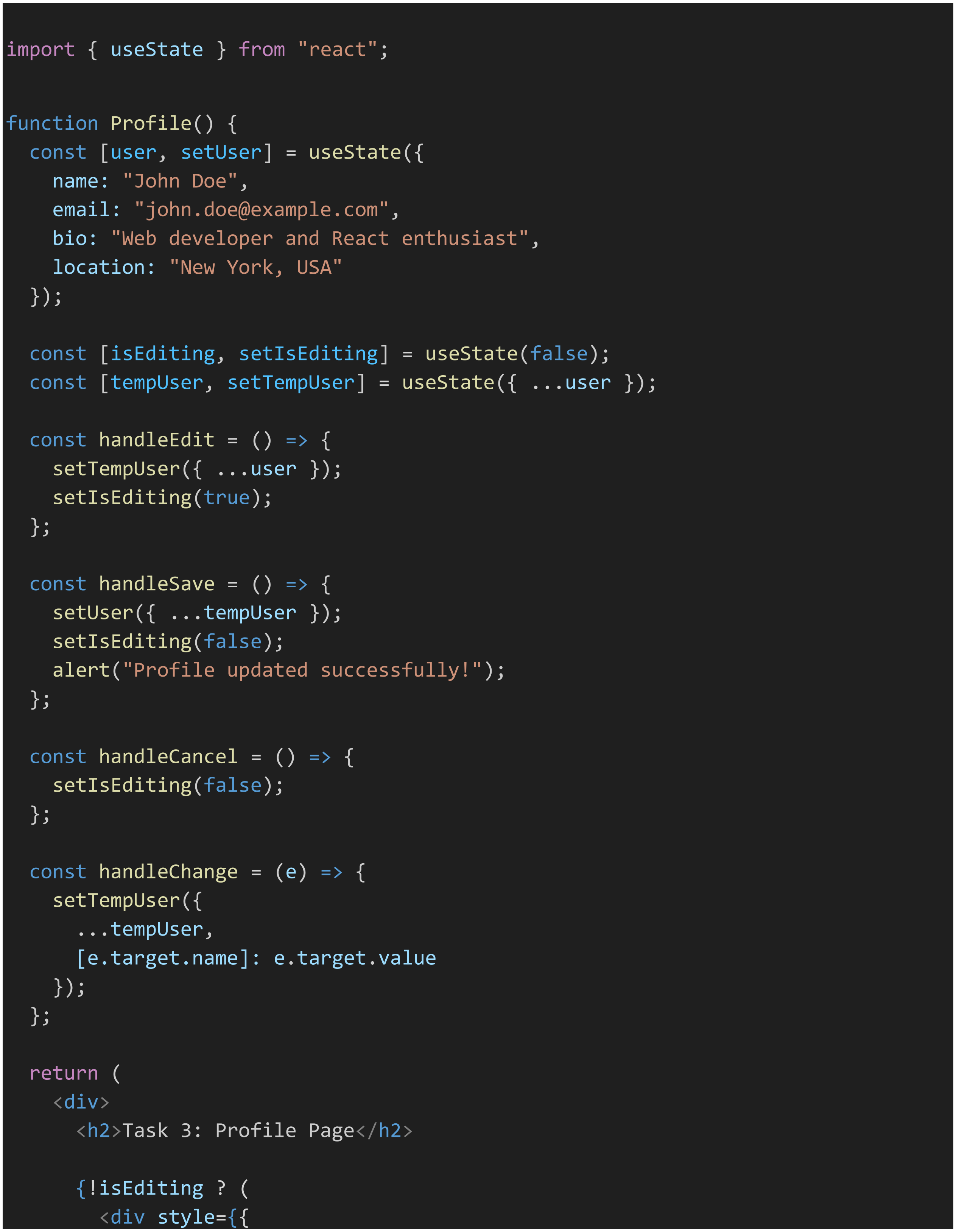
Quick Stats:

- Total Visits: 1,234
- Messages: 5
- Tasks Completed: 12

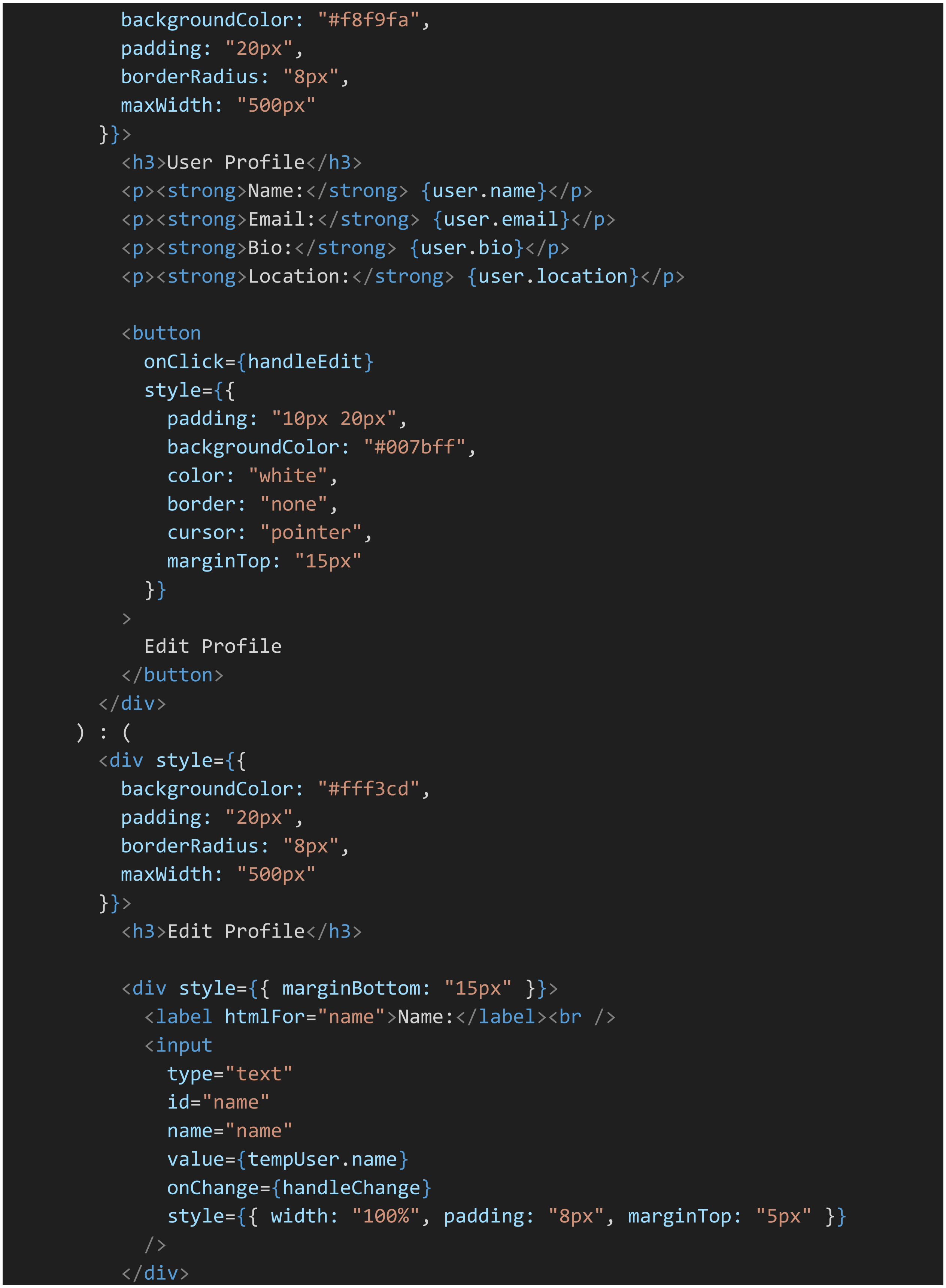
Navigation:

Go to Profile   Go to Home

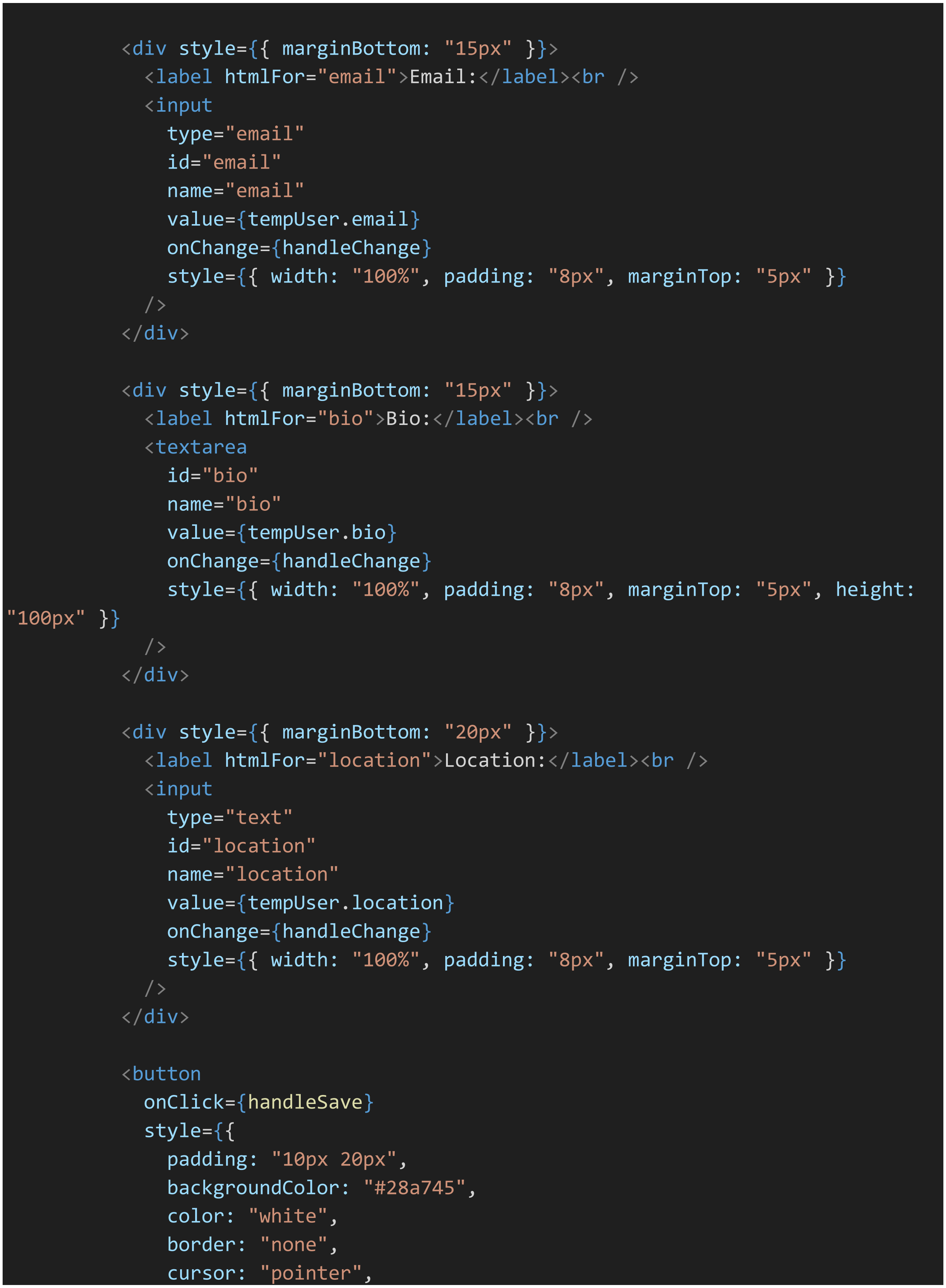
Profile



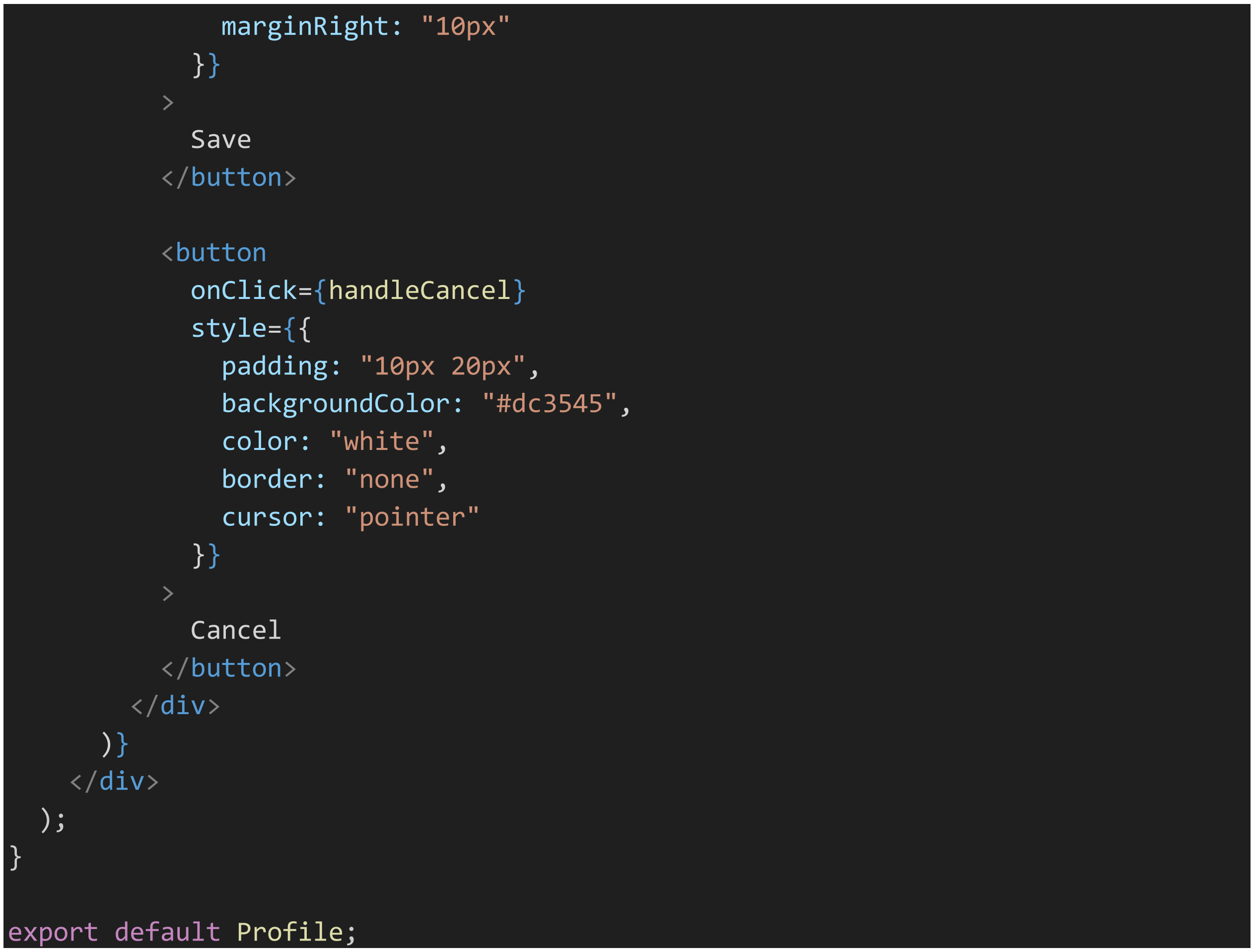
```
import { useState } from "react";

function Profile() {
  const [user, setUser] = useState({
    name: "John Doe",
    email: "john.doe@example.com",
    bio: "Web developer and React enthusiast",
    location: "New York, USA"
  });

  const [isEditing, setIsEditing] = useState(false);
  const [tempUser, setTempUser] = useState({ ...user });

  const handleEdit = () => {
    setTempUser({ ...user });
    setIsEditing(true);
  };

  const handleSave = () => {
    setUser({ ...tempUser });
    setIsEditing(false);
    alert("Profile updated successfully!");
  };

  const handleCancel = () => {
    setIsEditing(false);
  };

  const handleChange = (e) => {
    setTempUser({
      ...tempUser,
      [e.target.name]: e.target.value
    });
  };

  return (
    <div>
      <h2>Task 3: Profile Page</h2>

      {!isEditing ? (
        <div style={{
```



```
          backgroundColor: "#f8f9fa",
          padding: "20px",
          borderRadius: "8px",
          maxWidth: "500px"
        }}>
          <h3>User Profile</h3>
          <p><strong>Name:</strong> {user.name}</p>
          <p><strong>Email:</strong> {user.email}</p>
          <p><strong>Bio:</strong> {user.bio}</p>
          <p><strong>Location:</strong> {user.location}</p>

          <button
            onClick={handleEdit}
            style={{
              padding: "10px 20px",
              backgroundColor: "#007bff",
              color: "white",
              border: "none",
              cursor: "pointer",
              marginTop: "15px"
            }}
          >
            Edit Profile
          </button>
        </div>
      ) : (
        <div style={{
          backgroundColor: "#fff3cd",
          padding: "20px",
          borderRadius: "8px",
          maxWidth: "500px"
        }}>
          <h3>Edit Profile</h3>

          <div style={{ marginBottom: "15px" }}>
            <label htmlFor="name">Name:</label><br />
            <input
              type="text"
              id="name"
              name="name"
              value={tempUser.name}
              onChange={handleChange}
              style={{ width: "100%", padding: "8px", marginTop: "5px" }}
            />
          </div>
```

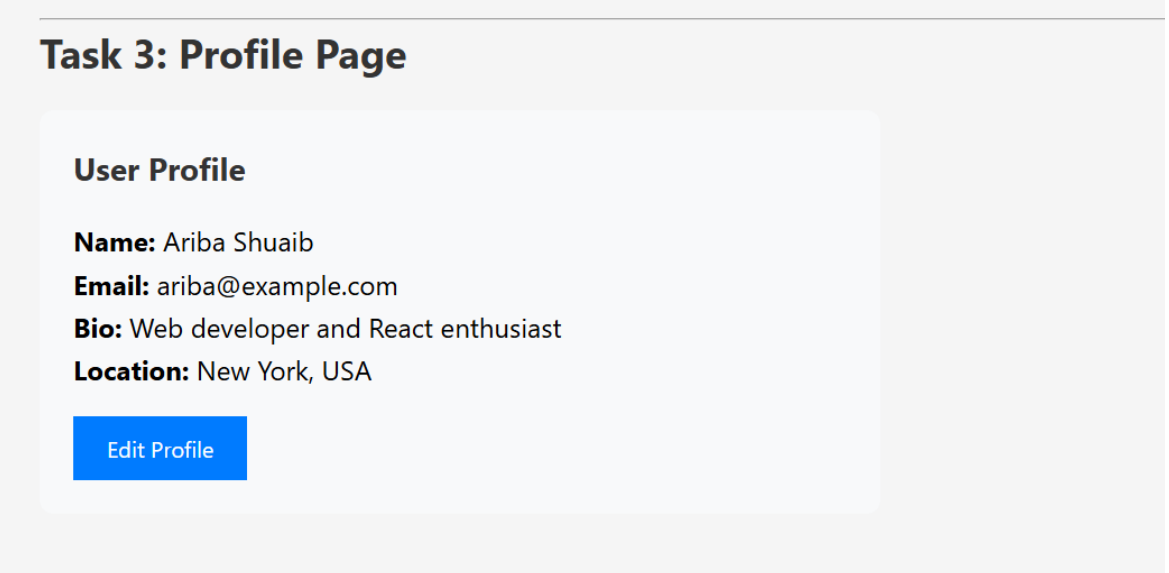


```
          <div style={{ marginBottom: "15px" }}>
            <label htmlFor="email">Email:</label><br />
            <input
              type="email"
              id="email"
              name="email"
              value={tempUser.email}
              onChange={handleChange}
              style={{ width: "100%", padding: "8px", marginTop: "5px" }}
            />
          </div>

          <div style={{ marginBottom: "15px" }}>
            <label htmlFor="bio">Bio:</label><br />
            <textarea
              id="bio"
              name="bio"
              value={tempUser.bio}
              onChange={handleChange}
              style={{ width: "100%", padding: "8px", marginTop: "5px", height:
"100px" }}
            />
          </div>

          <div style={{ marginBottom: "20px" }}>
            <label htmlFor="location">Location:</label><br />
            <input
              type="text"
              id="location"
              name="location"
              value={tempUser.location}
              onChange={handleChange}
              style={{ width: "100%", padding: "8px", marginTop: "5px" }}
            />
          </div>

          <button
            onClick={handleSave}
            style={{
              padding: "10px 20px",
              backgroundColor: "#28a745",
              color: "white",
              border: "none",
              cursor: "pointer",
```



```
              marginRight: "10px"
            }}
          >
            Save
          </button>

          <button
            onClick={handleCancel}
            style={{
              padding: "10px 20px",
              backgroundColor: "#dc3545",
              color: "white",
              border: "none",
              cursor: "pointer"
            }}
          >
            Cancel
          </button>
        </div>
      )}
    </div>
  );
}

export default Profile;
```

Output:



Task 3: Profile Page

User Profile

Name: Ariba Shuaib

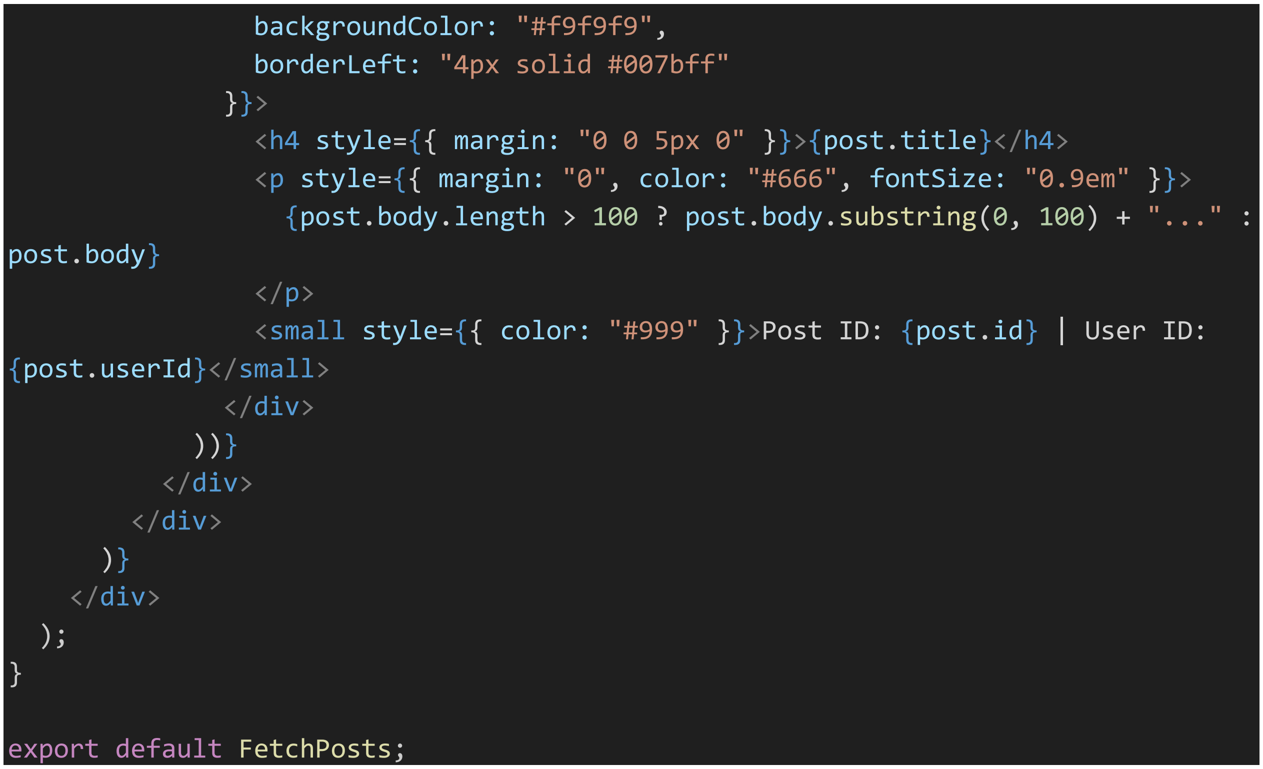
Email: ariba@example.com

Bio: Web developer and React enthusiast

Location: New York, USA

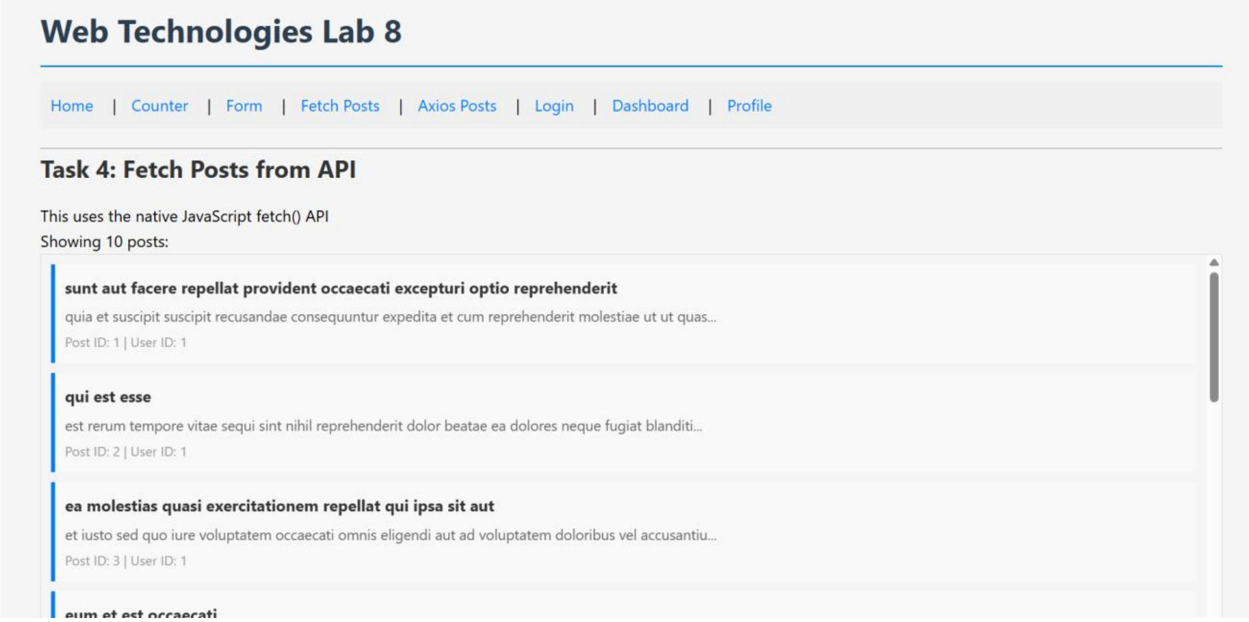
Task 4 — Fetch and display posts from a public API

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

function FetchPosts() {
  const [posts, setPosts] = useState([]);
  const [loading, setLoading] = useState(true);
  const [error, setError] = useState(null);

  useEffect(() => {
    setLoading(true);
    fetch("https://jsonplaceholder.typicode.com/posts")
      .then((response) => {
        if (!response.ok) {
          throw new Error("Failed to fetch posts");
        }
        return response.json();
      })
      .then((data) => {
        setPosts(data.slice(0, 10)); // Get first 10 posts
        setLoading(false);
      })
      .catch((err) => {
        setError(err.message);
        setLoading(false);
      });
  }, []);

  return (
    <div>
      <h2>Task 4: Fetch Posts from API</h2>
      <p>This uses the native JavaScript fetch() API</p>

      {loading && <p>Loading posts...</p>}
      {error && <p style={{ color: "red" }}>Error: {error}</p>}

      {!loading && !error && (
        <div>
          <p>Showing {posts.length} posts:</p>
          <div style={{ maxHeight: "400px", overflowY: "auto", border: "1px solid
#ddd", padding: "10px" }}>
            {posts.map((post) => (
              <div key={post.id} style={{
                padding: "10px",
                marginBottom: "10px",
```



```
                backgroundColor: "#f9f9f9",
                borderLeft: "4px solid #007bff"
              }}>
                <h4 style={{ margin: "0 0 5px 0" }}>{post.title}</h4>
                <p style={{ margin: "0", color: "#666", fontSize: "0.9em" }}>
                  {post.body.length > 100 ? post.body.substring(0, 100) + "..." :
post.body}
                </p>
                <small style={{ color: "#999" }}>Post ID: {post.id} | User ID:
{post.userId}</small>
              </div>
            ))}
          </div>
        </div>
      )}
    </div>
  );
}

export default FetchPosts;
```

Output:



Web Technologies Lab 8

Home | Counter | Form | Fetch Posts | Axios Posts | Login | Dashboard | Profile

Task 4: Fetch Posts from API

This uses the native JavaScript fetch() API

Showing 10 posts:

sunt aut facere repellat provident occaecati excepturi optio reprehenderit

quia et suscipit suscipit recusandae consequuntur expedita et cum reprehenderit molestiae ut ut quas...

Post ID: 1 | User ID: 1

qui est esse

est rerum tempore vitae sequi sint nihil reprehenderit dolor beatae ea dolores neque fugiat blanditi...

Post ID: 2 | User ID: 1

ea molestias quasi exercitationem repellat qui ipsa sit aut

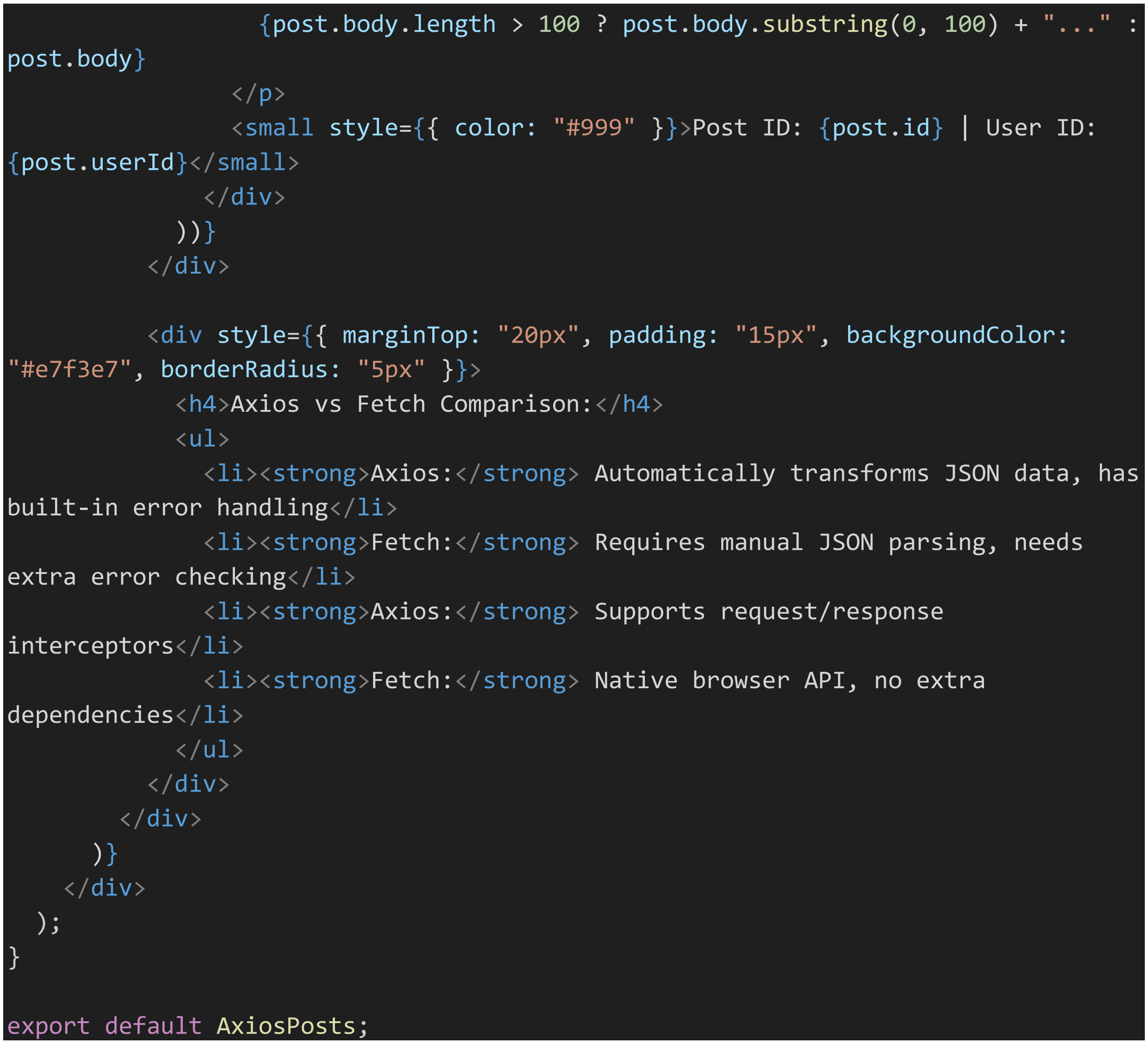
et iusto sed quo iure voluptatem occaecati omnis eligendi aut ad voluptatem doloribus vel accusantiu...

Post ID: 3 | User ID: 1

eum et est occaecati

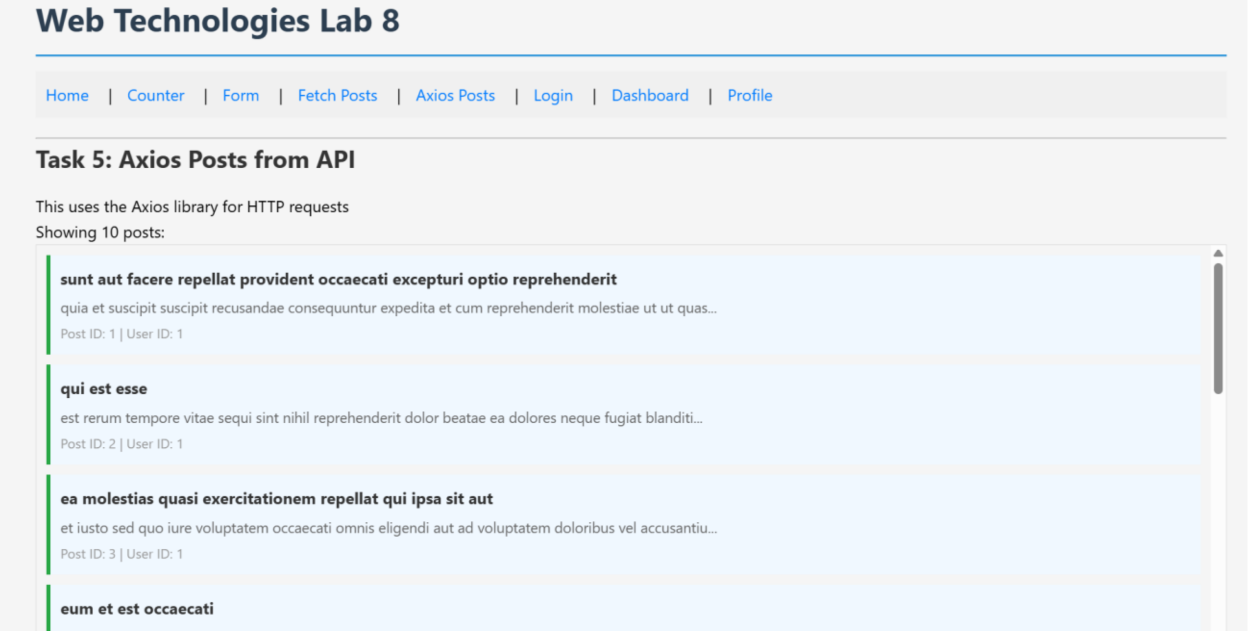
Task 5 — Convert API calls from fetch to axios

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

function AxiosPosts() {
  const [posts, setPosts] = useState([]);
  const [loading, setLoading] = useState(true);
  const [error, setError] = useState(null);

  useEffect(() => {
    setLoading(true);
    axios.get("https://jsonplaceholder.typicode.com/posts")
      .then((response) => {
        setPosts(response.data.slice(0, 10)); // Get first 10 posts
        setLoading(false);
      })
      .catch((err) => {
        setError(err.message);
        setLoading(false);
      });
  }, []);

  return (
    <div>
      <h2>Task 5: Axios Posts from API</h2>
      <p>This uses the Axios library for HTTP requests</p>

      {loading && <p>Loading posts...</p>}
      {error && <p style={{ color: "red" }}>Error: {error}</p>}

      {!loading && !error && (
        <div>
          <p>Showing {posts.length} posts:</p>
          <div style={{ maxHeight: "400px", overflowY: "auto", border: "1px solid #ddd", padding: "10px" }}>
            {posts.map((post) => (
              <div key={post.id} style={{
                padding: "10px",
                marginBottom: "10px",
                backgroundColor: "#f0f8ff",
                borderLeft: "4px solid #28a745"
              }}>
                <h4 style={{ margin: "0 0 5px 0" }}>{post.title}</h4>
                <p style={{ margin: "0", color: "#666", fontSize: "0.9em" }}>
```



```
                  {post.body.length > 100 ? post.body.substring(0, 100) + "..." :
post.body}
                </p>
                <small style={{ color: "#999" }}>Post ID: {post.id} | User ID:
{post.userId}</small>
              </div>
            ))}
          </div>

          <div style={{ marginTop: "20px", padding: "15px", backgroundColor:
"#e7f3e7", borderRadius: "5px" }}>
            <h4>Axios vs Fetch Comparison:</h4>
            <ul>
              <li><strong>Axios:</strong> Automatically transforms JSON data, has
built-in error handling</li>
              <li><strong>Fetch:</strong> Requires manual JSON parsing, needs
extra error checking</li>
              <li><strong>Axios:</strong> Supports request/response
interceptors</li>
              <li><strong>Fetch:</strong> Native browser API, no extra
dependencies</li>
            </ul>
          </div>
        </div>
      )}
    </div>
  );
}

export default AxiosPosts;
```

Output:



Web Technologies Lab 8

Home | Counter | Form | Fetch Posts | Axios Posts | Login | Dashboard | Profile

Task 5: Axios Posts from API

This uses the Axios library for HTTP requests

Showing 10 posts:

sunt aut facere repellat provident occaecati excepturi optio reprehenderit

quia et suscipit suscipit recusandae consequuntur expedita et cum reprehenderit molestiae ut ut quas...

Post ID: 1 | User ID: 1

qui est esse

est rerum tempore vitae sequi sint nihil reprehenderit dolor beatae ea dolores neque fugiat blanditi...

Post ID: 2 | User ID: 1

ea molestias quasi exercitationem repellat qui ipsa sit aut

et iusto sed quo iure voluptatem occaecati omnis eligendi aut ad voluptatem doloribus vel accusantiu...

Post ID: 3 | User ID: 1

eum et est occaecati