

## REACTJS (<https://reactjs.org/docs/getting-started.html>)

You'll need to have Node >= 14.00 and npm >= 5.6 on your computer

To create a project, run in your terminal or command prompt,

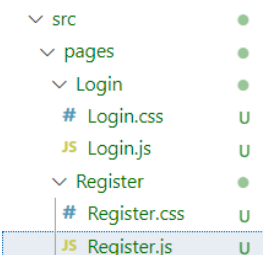
1. `npx create-react-app my-app`
2. `cd my-app`
3. `npm start`

\*my-app is the application name of your project

Create a new folder and name it as **pages** inside **src** folder.

Create the following files and folder inside **src/pages** folder,

1. **Login** (folder)
  - a. **Login.js**
  - b. **Login.css**
2. **Register** (folder)
  - a. **Register.js**
  - b. **Register.css**



src	
pages	
Login	
Login.css	U
Login.js	U
Register	
Register.css	U
Register.js	U

### Login.js

```
1. import './Login.css';
2.
3. function Login() {
4.   return (
5.     <div className="Login">
6.       Login page
7.     </div>
8.   );
9. }
10.
11. export default Login;
```

### Register.js

```
1. import './Register.css';
2.
3. function Register() {
4.   return (
5.     <div className="Register">
6.       Register page
7.     </div>
8.   );
9. }
10.
11. export default Register;
```

To use the **Login component** or **Register component**, you must import the component (see line 2) and you can use it by using the **<Login/>** or **<Register/>** tags.

Modify **App.js**

```
1. import './App.css';
2. import Login from './pages/Login/Login';
3.
4. function App() {
5.   return (
6.     <div className="App">
7.       <Login/>
8.     </div>
9.   );
10. }
11.
12. export default App;
```

Create a login page by modifying the **Login.css** and **Login.js** for styles. Install **React Hook Form** (<https://react-hook-form.com/get-started>) or run the npm command using your command prompt (make sure that it is inside your project) **npm install react-hook-form**

## Login.css

```
1. .main-container {
2.   display: flex;
3.   flex-direction: column;
4.   justify-content: center;
5.   align-items: center;
6.   height: 100vh;
7. }
8.
9. .form-container {
10.  padding: 4rem 2rem;
11.  border: 1px solid #ccc;
12.  border-radius: 0.5rem;
13.  display: flex;
14.  flex-direction: column;
15.  justify-content: center;
16.  align-items: center;
17. }
18. .form-container > div {
19.  margin-bottom: 12px;
20. }
21.
22. .form-group {
23.  display: flex;
24. }
25.
26. .form-group label {
27.  width: 70px;
28.  margin-right: 8px;
29.  text-align: left;
30. }
31. .errors {
32.  color: red;
33.  font-size: 12px;
34. }
35. .submit-container {
36.  margin-top: 12px;
37.  text-align: center;
38. }
39. .register-container {
40.  text-align: center;
41. }
```

## Login.js

```
1. import { useForm } from "react-hook-form";
2. import "./Login.css";
3.
4. function Login() {
5.   const {
6.     register,
7.     handleSubmit,
8.     formState: { errors },
9.   } = useForm({
10.    defaultValues: {
11.      email: "",
12.      password: "",
13.    },
14.  });
15.  const onSubmit = (data) => {
16.    console.log(data, apiResponse);
17.  };
18.  return (
19.    <div className="Login">
20.      <div className="main-container">
21.        <h3>Login</h3>
22.        <form onSubmit={handleSubmit(onSubmit)}>
23.          <div className="form-container">
24.            <div>
25.              <div className="form-group">
26.                <label>E-mail:</label>
27.                <input type="text" {...register("email", { required: true })} />
28.              </div>
29.              {errors.email && (
30.                <span className="errors">This field is required</span>
31.              )}
32.            </div>
33.            <div>
34.              <div className="form-group">
35.                <label>Password:</label>
36.                <input
37.                  type="password"
38.                  {...register("password", { required: true })}
39.                />
40.              </div>
41.              {errors.password && (
42.                <span className="errors">This field is required</span>
43.              )}
44.            </div>
45.            <div className="submit-container">
46.              <button type="submit">Login</button>
47.            </div>
48.            <div className="register-container">
49.              <a>
50.                <small>Register</small>
51.              </a>
52.            </div>
53.          </div>
54.        </form>
55.      </div>
56.    </div>
57.  );
58. }
59.
60. export default Login;
```

To submit the data to the API, install the **axios** (<https://axios-http.com/docs/intro>) or run using your cmd `npm install axios`.

Modify **Login.js**, import axios to your component by placing it below `import './Login.css'`.

```
3. import axios from "axios";
```

---

Hook the API login request using axios inside of onSubmit function.

```
26. const onSubmit = async (data) => {
27.   await axios({
28.     method: "post",
29.     url: "user/login",
30.     data,
31.     headers: { "Access-Control-Allow-Origin": "*" },
32.   })
33.   .then((res) => {
34.     localStorage.setItem("accessToken", res.data.access_token);
35.     navigate("/main/dashboard");
36.   })
37.   .catch((e) => {
38.     alert(e.response.data.message);
39.   });
40. };
```

To navigate from Login page to Register page, install **React Router** by running

`npm install react-router-dom`.

Modify **Login.js** and add `href="/register"` to **anchor tag**

```
63. <div className="register-container">
64.   <a href="/register">
65.     <small>Register</small>
66.   </a>
67. </div>
```

Modify **App.js**, we need to import the `createBrowserRouter and Router` from react-router-dom. Import also the **Login component (Login.js)** and **Register component (Register.js)**.

## App.js

```
1. import './App.css';
2. import { createBrowserRouter, RouterProvider } from "react-router-dom";
3. import Login from './pages/Login/Login';
4. import Register from './pages/Register/Register';
5.
6. const router = createBrowserRouter([
7.   {
8.     path: "/",
9.     element: <Login />,
10.  },
11.  {
12.    path: "/register",
13.    element: <Register />,
14.  },
15. ]);
16.
17. function App() {
18.   return (
19.     <div className="App">
20.       <RouterProvider router={router} />
21.     </div>
22.   );
23. }
24.
25. export default App;
```

## Modify Register.js

```
1. import { useForm } from "react-hook-form";
2. import './Register.css';
3. import axios from "axios";
4. import { useNavigate } from "react-router-dom";
5.
6. function Register() {
7.   const navigate = useNavigate();
8.   const {
9.     register,
10.    handleSubmit,
11.    formState: { errors },
12.  } = useForm({
13.    defaultValues: {
14.      email: "",
15.      password: "",
16.      firstName: "",
17.      middleName: "",
18.      lastName: "",
19.      contactNo: "",
20.    },
21.  });
22.   const onSubmit = async (data) => {
23.     const apiResponse = await axios({
24.       method: "post",
25.       url: "user/register",
26.       data,
27.       headers: {
28.         "Access-Control-Allow-Origin": "*",
29.         "Access-Control-Allow-Credentials": true,
```

```

30.     },
31.   })
32.     .then((res) => {
33.       alert(res.data.message);
34.       navigate("/");
35.       return res;
36.     })
37.     .catch((e) => {
38.       alert(e.response.data.message);
39.     });
40.   });
41.   return (
42.     <div className="Register">
43.       <div className="main-container">
44.         <h3>Register</h3>
45.         <form onSubmit={handleSubmit(onSubmit)}>
46.           <div className="form-container">
47.             <div>
48.               <div className="form-group">
49.                 <label>E-mail:</label>
50.                 <input type="text" {...register("email", { required: true })} />
51.               </div>
52.               {errors.email && (
53.                 <span className="errors">This field is required</span>
54.               )}
55.             </div>
56.             <div>
57.               <div className="form-group">
58.                 <label>Password:</label>
59.                 <input
60.                   type="password"
61.                   {...register("password", { required: true })}
62.                 />
63.               </div>
64.               {errors.password && (
65.                 <span className="errors">This field is required</span>
66.               )}
67.             </div>
68.             <div>
69.               <div className="form-group">
70.                 <label>First name:</label>
71.                 <input
72.                   type="firstName"
73.                   {...register("firstName", { required: true })}
74.                 />
75.               </div>
76.               {errors.firstName && (
77.                 <span className="errors">This field is required</span>
78.               )}
79.             </div>
80.             <div>
81.               <div className="form-group">
82.                 <label>Middle name:</label>
83.                 <input
84.                   type="middleName"
85.                   {...register("middleName", { required: true })}
86.                 />
87.               </div>
88.               {errors.middleName && (

```

```

91.         <span className="errors">This field is required</span>
92.     )}
93. </div>
94.
95.     <div>
96.         <div className="form-group">
97.             <label>Last name:</label>
98.             <input
99.                 type="lastName"
100.                 {...register("lastName", { required: true })}
101.             />
102.         </div>
103.         {errors.lastName && (
104.             <span className="errors">This field is required</span>
105.         )}
106.     </div>
107.
108.     <div>
109.         <div className="form-group">
110.             <label>Contact no:</label>
111.             <input
112.                 type="contactNo"
113.                 {...register("contactNo", { required: true })}
114.             />
115.         </div>
116.         {errors.contactNo && (
117.             <span className="errors">This field is required</span>
118.         )}
119.     </div>
120.
121.     <div className="submit-container">
122.         <button type="submit">Register</button>
123.     </div>
124.     <div className="login-container">
125.         <a href="/">
126.             <small>Back to Login</small>
127.         </a>
128.     </div>
129. </div>
130. </form>
131. </div>
132. </div>
133. );
134. }
135.
136. export default Register;

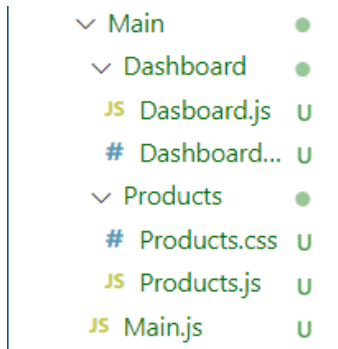
```



## Register.css

```
1. .main-container {
2.   display: flex;
3.   flex-direction: column;
4.   justify-content: center;
5.   align-items: center;
6.   height: 100vh;
7. }
8.
9. .Register .form-container {
10.  width: 360px;
11.  padding: 4rem 2rem;
12.  border: 1px solid #ccc;
13.  border-radius: 0.5rem;
14.  display: flex;
15.  flex-direction: column;
16.  align-items: stretch;
17. }
18. .Register .form-container > div {
19.  margin-bottom: 12px;
20. }
21.
22. .Register .form-group {
23.  display: flex;
24. }
25.
26. .Register .form-group label {
27.  flex-grow: 1;
28.  margin-right: 8px;
29.  text-align: left;
30. }
31. .Register .form-group input {
32.  flex-grow: 2;
33. }
34.
35. .errors {
36.  color: red;
37.  font-size: 12px;
38. }
39. .submit-container {
40.  margin-top: 12px;
41.  text-align: center;
42. }
43. .login-container {
44.  text-align: center;
45. }
```

Create a folder, js and css with filename **Main** inside **pages** folder. Inside **Main** folder, create a new component (folder, js and css files) for **Dashboard** and **Products**



### Main.css

```
1. .Main {
2.   display: flex;
3.   overflow: hidden;
4.   max-height: 100vh;
5. }
6. .Main .container {
7.   display: flex;
8.   width: 100vw;
9. }
10. .Main .container .outlet {
11.   flex-grow: 3;
12.   padding: 1rem;
13. }
14. .Main .container .navigation {
15.   background-color: aliceblue;
16.   padding-top: 40px;
17.   height: 100vh;
18. }
19. .Main .container .navigation ul {
20.   list-style: none;
21.   overflow: hidden;
22.   padding: 0;
23.   margin: 0;
24. }
25.
26. .Main .container .navigation ul li {
27.   padding: 10px 80px 10px 20px;
28. }
29. .Main .container .navigation ul li.active,
30. .Main .container .navigation ul li:hover {
31.   background: #ccc;
32. }
33. .Main .container .navigation ul li a {
34.   text-decoration: none;
35. }
36.
37. .Main .container .navigation ul li.logout {
38.   cursor: pointer;
39. }
```

## Main.js

```
1. import { useEffect } from "react";
2. import { Outlet, useNavigate } from "react-router-dom";
3. import "./Main.css";
4.
5. function Main() {
6.   const accessToken = localStorage.getItem("accessToken");
7.   const navigate = useNavigate();
8.   const handleLogout = () => {
9.     localStorage.removeItem("accessToken");
10.    navigate("/");
11.  };
12.
13.  useEffect(() => {
14.    if (
15.      accessToken === undefined ||
16.      accessToken === "" ||
17.      accessToken === null
18.    ) {
19.      handleLogout();
20.    }
21.  }, []);
22.  return (
23.    <div className="Main">
24.      <div className="container">
25.        <div className="navigation">
26.          <ul>
27.            <li>
28.              <a href="/main/dashboard">Dashboard</a>
29.            </li>
30.            <li>
31.              <a href="/main/products">Products</a>
32.            </li>
33.            <li className="logout">
34.              <a onClick={handleLogout}>Logout</a>
35.            </li>
36.          </ul>
37.        </div>
38.        <div className="outlet">
39.          <Outlet />
40.        </div>
41.      </div>
42.    </div>
43.  );
44. }
45.
```

## Dashboard.js

```
1. import "../Dashboard.css";
2.
3. function Dashboard() {
4.   return (
5.     <div className="Dashboard">
6.       <h1>Dashboard</h1>
7.       <div className="content">Welcome!</div>
8.     </div>
9.   );
10. }
11.
12. export default Dashboard;
```

## Products.js

```
1. import "../Products.css";
2.
3. function Products() {
4.   return (
5.     <div className="Products">
6.       <h1>Products</h1>
7.       <div className="content">Products page</div>
8.     </div>
9.   );
10. }
11.
12. export default Products;
```

Modify **App.js** and add the **Main** component to the router

```
9. const router = createBrowserRouter([
10.   {
11.     path: "/",
12.     element: <Login />,
13.   },
14.   {
15.     path: "/register",
16.     element: <Register />,
17.   },
18.   {
19.     path: "/main",
20.     element: <Main />,
21.     children: [
22.       {
23.         path: "/main/dashboard",
24.         element: <Dashboard />,
25.       },
26.       {
27.         path: "/main/products",
28.         element: <Products />,
29.       },
30.     ],
31.   },
32. ]);
```