

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
КРЕМЕНЧУЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ МИХАЙЛА ОСТРОГРАДСЬКОГО  
Навчально-науковий інститут електричної інженерії  
та інформаційних технологій  
КАФЕДРА АВТОМАТИЗАЦІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

ЗВІТ

З ЛАБОРАТОРНИХ РОБІТ  
З НАВЧАЛЬНОЇ ДИСЦИПЛІНИ  
«Frontend-розробка»

Виконав студент групи КН-23-1

Полинько Ігор Миколайович

Перевірив старший викладач кафедри АІС Притчин О. С.

КРЕМЕНЧУК 2025

## Лабораторна робота № 7

**Тема:** Робота з навігацією в React

**Мета:** Познайомитись і отримати головні навички роботи із навігацією між сторінками в React додатку

### Виконання завдання лабораторної роботи:

Створіть проєкт React. Додаток має містити 8 сторінок де виводитиметься певна інформація.

За допомогою react router налаштуйте наступну навігацію для створених сторінок. В додатку має бути домашня сторінка та ще 4 сторінки, що представлятимуть інші розділи на тому ж рівні. Користувач маж мати можливість вільно переходити між ними.

Одна з цих 4 сторінок (не домашня) має бути контейнером і містити 3 вкладені сторінки. Це будуть підрозділи. Між ними теж повинна бути налагоджена навігація.

Ще одна зі сторінок має містити версію, що зчитуватиме з url та виводитиме на екран значення певного параметра на кшталт id.

В цьому завданні не важливо наповнення сторінок. Важливо показати вміння створити і налагодити навігацію в React додатку.

### index.js:

```
import React from "react";
import ReactDOM from "react-dom/client";
import { BrowserRouter } from "react-router-dom";
import App from "./App";

const root = ReactDOM.createRoot(document.getElementById("root"));
root.render(
  <BrowserRouter>
    <App />
  </BrowserRouter>
);

// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
```

## Navbar.jsx:

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

export default function Navbar() {
  const active = ({ isActive }) => ({
    fontWeight: isActive ? "bold" : "normal",
    textDecoration: isActive ? "underline" : "none",
    marginRight: "15px"
  });

  return (
    <nav style={{ marginBottom: "20px" }}>
      <NavLink to="/" style={active}>Home</NavLink>
      <NavLink to="/about" style={active}>About</NavLink>
      <NavLink to="/contact" style={active}>Contact</NavLink>
      <NavLink to="/products" style={active}>Products</NavLink>
      <NavLink to="/dashboard" style={active}>Dashboard</NavLink>
    </nav>
  );
}
```

## App.jsx:

```
import { Routes, Route } from "react-router-dom";
import Navbar from "../Navbar";
import Home from "../pages/Home";
import About from "../pages/About";
import Contact from "../pages/Contact";
import Products from "../pages/Products";
import ProductDetail from "../pages/ProductDetail";
import DashboardLayout from "../pages/Dashboard/DashboardLayout";
import DashboardHome from "../pages/Dashboard/DashboardHome";
import DashboardProfile from "../pages/Dashboard/DashboardProfile";
import DashboardSettings from "../pages/Dashboard/DashboardSettings";

export default function App() {
  return (
    <div style={{ padding: "20px", fontFamily: "Arial" }}>
      <h2>Лабораторна робота №7</h2>
      <Navbar />
      <Routes>
        <Route path="/" element={<Home />} />
        <Route path="/about" element={<About />} />
        <Route path="/contact" element={<Contact />} />
        <Route path="/products" element={<Products />} />
        <Route path="/products/:id" element={<ProductDetail />} />

        <Route path="/dashboard" element={<DashboardLayout />}>
          <Route index element={<DashboardHome />} />
          <Route path="profile" element={<DashboardProfile />} />
          <Route path="settings" element={<DashboardSettings />} />
        </Route>
      </Routes>
    </div>
  );
}
```

```
        </Route>
      </Routes>
    </div>
  );
}
```

### Home.jsx:

```
export default function Home() {
  return <h3>Головна сторінка</h3>;
}
```

### About.jsx:

```
export default function About() {
  return <h3>Про нас</h3>;
}
```

### Contact.jsx:

```
export default function Contact() {
  return <h3>Контакти</h3>;
}
```

### Products.jsx:

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

export default function Products() {
  const navigate = useNavigate();
  const products = [1, 2, 3];

  return (
    <div>
      <h3>Список продуктів</h3>
      <ul>
        {products.map(id => (
          <li key={id}>
            <Link to={` /products/${id}`}>Переглянути продукт #{id}</Link>
          </li>
        ))}
      </ul>
      <button onClick={() => navigate("/")}>Повернутись на головну</button>
    </div>
  );
}
```

### ProductDetail.jsx:

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

export default function ProductDetail() {
  const { id } = useParams();
  return <h3>Деталі продукту: ID = {id}</h3>;
}
```

### DashboardLayout.jsx:

```
import { NavLink, Outlet } from "react-router-dom";

export default function DashboardLayout() {
  return (
    <div>
      <h3>Панель керування</h3>
      <nav>
        <NavLink to="/dashboard" end style={{ marginRight: 10 }}>Home</NavLink>
        <NavLink to="/dashboard/profile" style={{ marginRight: 10 }}>Profile</NavLink>
        <NavLink to="/dashboard/settings">Settings</NavLink>
      </nav>
      <hr />
      <Outlet />
    </div>
  );
}
```

### DashboardHome.jsx:

```
export default function DashboardHome() {
  return <p>Головна сторінка Dashboard</p>;
}
```

### DashboardProfile.jsx:

```
export default function DashboardProfile() {
  return <p>Профіль користувача</p>;
}
```

### DashboardSettings.jsx:

```
export default function DashboardSettings() {
  return <p>Налаштування акаунта</p>;
}
```

Результат:

## Лабораторна робота №7

[Home](#) [About](#) [Contact](#) [Products](#) [Dashboard](#)

### Головна сторінка

Рисунок 7.1 – Головна сторінка

## Лабораторна робота №7

[Home](#) [About](#) [Contact](#) [Products](#) [Dashboard](#)

### Про нас

Рисунок 7.2 – Сторінка «Про нас»

## Лабораторна робота №7

[Home](#) [About](#) [Contact](#) [Products](#) [Dashboard](#)

### Контакти

Рисунок 7.3 – Сторінка «Контакти»

## Лабораторна робота №7

[Home](#) [About](#) [Contact](#) [Products](#) [Dashboard](#)

### Список продуктів

- [Переглянути продукт #1](#)
- [Переглянути продукт #2](#)
- [Переглянути продукт #3](#)

[Повернутись на головну](#)

Рисунок 7.4 – Сторінка «Продукти»

## Лабораторна робота №7

[Home](#) [About](#) [Contact](#) [Products](#) [Dashboard](#)

Деталі продукту: ID = 1

Рисунок 7.5 – Деталі продукту #1

## Лабораторна робота №7

[Home](#) [About](#) [Contact](#) [Products](#) [Dashboard](#)

Деталі продукту: ID = 2

Рисунок 7.6 – Деталі продукту #2

## Лабораторна робота №7

[Home](#) [About](#) [Contact](#) [Products](#) [Dashboard](#)

Деталі продукту: ID = 3

Рисунок 7.7 – Деталі продукту #3

## Лабораторна робота №7

[Home](#) [About](#) [Contact](#) [Products](#) [Dashboard](#)

Панель керування

[Home](#) [Profile](#) [Settings](#)

---

Головна сторінка Dashboard

Рисунок 7.8 – Головна сторінка Dashboard

## Лабораторна робота №7

[Home](#) [About](#) [Contact](#) [Products](#) [Dashboard](#)

### Панель керування

[Home](#) [Profile](#) [Settings](#)

---

Профіль користувача

Рисунок 7.9 – Сторінка профілю користувача Dashboard

## Лабораторна робота №7

[Home](#) [About](#) [Contact](#) [Products](#) [Dashboard](#)

### Панель керування

[Home](#) [Profile](#) [Settings](#)

---

Налаштування акаунта

Рисунок 7.10 – Сторінка налаштування акаунта Dashboard

**Висновок:** на цій лабораторній роботі ми познайомились і отримали головні навички роботи із навігацією між сторінками в React додатку.

### Контрольні питання:

#### 1. Основні компоненти React Router

- 1) BrowserRouter / HashRouter – компоненти-обгортки, які забезпечують маршрутизацію в додатку. Використовуються один раз на рівні всього застосунку.
- 2) Routes – контейнер для всіх маршрутів (Route).
- 3) Route – визначає окремий маршрут з вказанням шляху (path) та компонента для відображення (element).
- 4) Link / NavLink – компоненти для переходу між маршрутами без перезавантаження сторінки.
- 5) Outlet – точка рендерингу для вкладених маршрутів.



6) `useNavigate`, `useParams`, `useLocation`, `useSearchParams` – хуки для програмної навігації та отримання параметрів маршруту.

## 2. Різниця між `Routes` та `Route`

– `Routes` – контейнер для всіх маршрутів. Використовується для групування маршрутів і замінює застарілий `Switch`.

– `Route` – компонент, який описує окремий маршрут, вказує шлях (`path`) та компонент для відображення (`element`).

### Приклад використання:

```
<Routes>
  <Route path="/" element={<Home />} />
  <Route path="/about" element={<About />} />
</Routes>
```

## 3. Відмінність `NavLink` від `Link`

– `Link` – стандартне навігаційне посилання для переходу між сторінками.

– `NavLink` – аналог `Link`, але дозволяє автоматично визначати активний маршрут та застосовувати відповідний CSS-клас або стиль.

### Приклад:

```
<NavLink to="/home" className={({ isActive }) => isActive ? "active" : ""}>
  Home
</NavLink>
```

## 4. Програмна передача параметрів через маршрут

Для програмної навігації використовується хук `useNavigate()`:

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

```
const Example = () => {
  const navigate = useNavigate();
  navigate("/user/42"); // передача параметра "42" через URL
}
```

```
};
```

## 5. Отримання параметрів маршруту

Для отримання параметрів поточного маршруту використовується хук `useParams()`:

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

```
const User = () => {  
  const { id } = useParams();  
  return <h2>User ID: {id}</h2>;  
};
```

## 6. Параметр `index` у вкладених маршрутах

Параметр `index` позначає маршрут за замовчуванням у вкладених маршрутах. Він використовується для відображення компонента при переході на батьківський шлях, якщо не вказаний дочірній маршрут.

### Приклад:

```
<Route path="dashboard" element={<Dashboard />}>  
  <Route index element={<DashboardHome />} /> // маршрут за замовчуванням  
  <Route path="settings" element={<Settings />} />  
</Route>
```