

Tworzenie podstron

Do utworzenia podstron potrzebne jest zainstalowanie biblioteki ReactRouter, która zawiera komponenty umożliwiające tworzenie podstron.

npm install react-router-dom

Trzeba utworzyć plik z nawigacją np. Navigation.js (pliki muszą zaczynać się od wielkiej litery!)

```
import { Link } from "react-router-dom";
import 'bootstrap/dist/css/bootstrap.css';
import 'bootstrap/dist/js/bootstrap.bundle.min.js';

export default function Navigation(){
  return(
    <ul className="nav justify-content-center">
      <li className="nav-item">
        <Link className="nav-link" to="/">Home</Link>
      </li>
      <li className="nav-item dropdown">
        <a className="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown">
          Opis
        </a>
        <ul className="dropdown-menu">
          <li>
            <Link className="dropdown-item" to="/opis1">Opis1</Link>
          </li>
          <li>
            <Link className="dropdown-item" to="/opis2">Opis2</Link>
          </li>
          <li>
            <Link className="dropdown-item" to="/opis3">Opis3</Link>
          </li>
        </ul>
      </li>
      <li className="nav-item">
        <Link className="nav-link" to="/zdjecia">Zdjecia</Link>
      </li>
    </ul>
  )
}
```

Komponent **Link** odpowiada za nawigację do innej podstrony, parametr **to** wskazuje na adres podstrony, czyli np.

```
<Link className="dropdown-item" to="/opis1">Opis1</Link>
```

mówi, że jeżeli ktoś kliknie w Opis1 ma przejść do /opis1

w Navigation.js jest też pokazane jak to można ostylewać za pomocą bootstrap (pamiętaj, żeby go wcześniej zainstalować). I na górze zaimportować

```
import 'bootstrap/dist/css/bootstrap.css';  
import 'bootstrap/dist/js/bootstrap.bundle.min.js';  
bootstrap.bundle.min.js; jest potrzebne do menu rozwijalnego, jeżeli takiego nie robisz, nie musisz importować.
```

plik Opis1.js wygląda tak (oczywiście można tu tworzyć swoje funkcje, cała wygląd podstrony):

```
export default function Opis1(){  
  return(  
    <div>  
      <h3>Strona z opisem 1</h3>  
    </div>  
  )  
}
```

Plik App.js

```
import './App.css';  
import Navigation from './Navigation';  
import Home from './Home';  
import Opis1 from './Opis1';  
import Opis2 from './Opis2';  
import Opis3 from './Opis3';  
import Zdjecia from './Zdjecia';  
import { BrowserRouter as Router,Routes,Route } from 'react-router-dom';  
//import 'bootstrap/dist/js/bootstrap.bundle.min.js';
```

```
function App() {  
  return (  
    <Router>  
    <div className="App">  
  
    <Navigation/>  
    <Routes>  
      <Route path="/" element={<Home/>}/>  
      <Route exact path="/opis1" element={<Opis1/>}/>  
      <Route exact path="/opis2" element={<Opis2/>}/>  
      <Route exact path="/opis3" element={<Opis3/>}/>  
      <Route exact path="/zdjecia" element={<Zdjecia/>}/>  
    <Route path="*" element={<NotFound/>}/>  
    </Routes>  
  )  
}
```

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

```
export default App;
```

trzeba najpierw zaimportować wszystkie komponenty, np.

```
import Navigation from './Navigation';
import Home from './Home';
import Opis1 from './Opis1';
```

oraz pakiety z ReactRouter

```
import { BrowserRouter as Router, Routes, Route } from 'react-router-dom';
```

BrowserRouter w tym przypadku ma alias **Router**, ale nie musisz tego robić. On włącza routing.

```
<Navigation/>
```

W tym miejscu wstawiamy komponent `Navigation.js`

Routes sprawdza aktualny adres URL i uruchamia odpowiednią podstronę

```
<Route exact path='/opis1' element={<Opis1/>}/>
```

exact oznacza, że adres URL musi być dokładnie taki sam, jak w parametrze **path**, gdybyśmy tego nie wpisali, to każdy nie pasujący adres URL będzie pasował do „/”

Na samym końcu piszemy

```
<Route path='*' element={<NotFound/>}/>
```

ta linijka wskazuje na to co ma się pojawić, gdy ktoś próbuje wyświetlić stronę, której nie ma