

## Zadanie 1 „useContext”

Hook `useContext` w React służy do **pobierania wartości z kontekstu** (czyli z obiektu `Context` utworzonego przez `React.createContext`) **bez konieczności ręcznego przekazywania propsów przez kolejne komponenty (prop drilling)**.

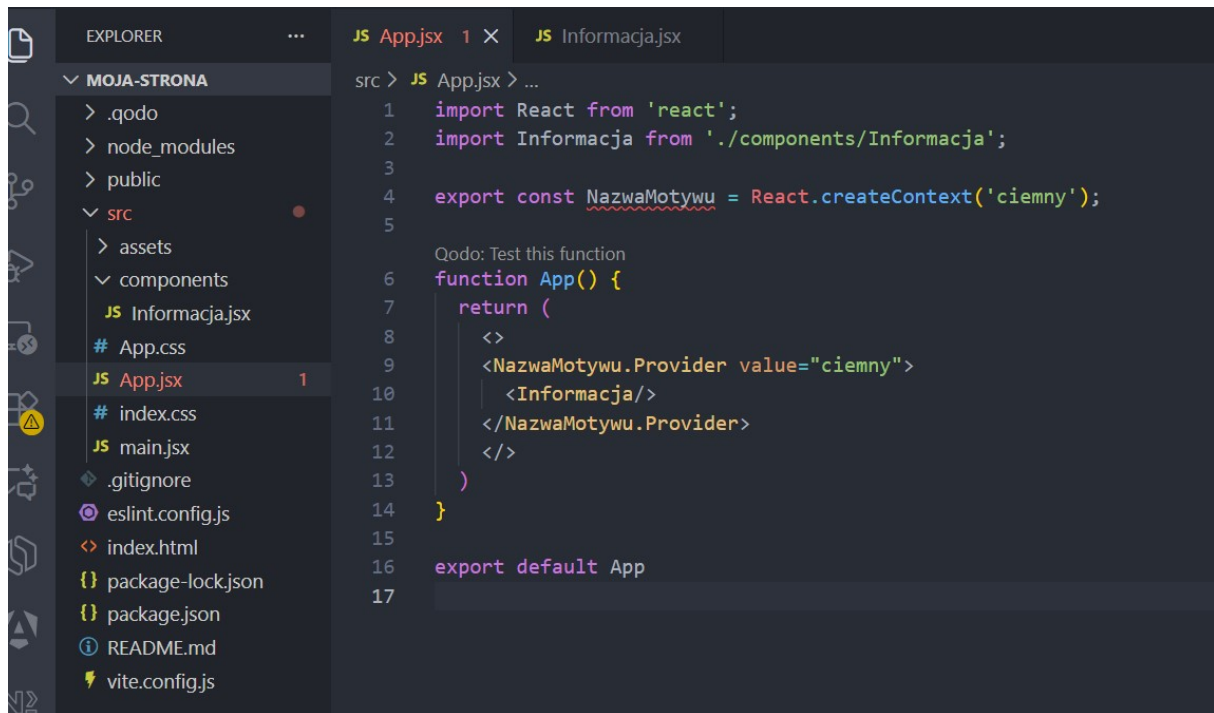
### Po co stosuje się `useContext`?

Kiedy kilka komponentów w różnych miejscach drzewa ma korzystać z tych samych danych (np. język aplikacji, motyw, dane użytkownika), zamiast przekazywać je przez propsy na każdej warstwie, używamy **Context API**.

Wykonaj Ćwiczenie:

Utwórz nowy projekt w React i nazwij go StarWars.

Projekt posiada jeden komponent „Informacja” który używa `useContext` utworzony w `App.jsx`



The screenshot shows a code editor with a file explorer on the left and two code files open. The file explorer shows a project structure with a `src` directory containing `assets`, `components`, and `Informacja.jsx`. The `App.jsx` file is selected, showing the following code:

```
src > JS App.jsx > ...
1  import React from 'react';
2  import Informacja from './components/Informacja';
3
4  export const NazwaMotywu = React.createContext('ciemny');
5
6  Qodo: Test this function
7  function App() {
8    return (
9      <>
10       <NazwaMotywu.Provider value="ciemny">
11         <Informacja/>
12       </NazwaMotywu.Provider>
13     </>
14   )
15 }
16
17 export default App
```

The screenshot shows a VS Code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with a 'src' directory containing 'Informacja.js'. The code editor shows the implementation of the 'Informacja' component in 'Informacja.js'. It imports 'useContext' from 'react' and 'NazwaMotywu' from './App'. The component function 'Informacja' uses 'useContext(NazwaMotywu)' to get the current theme and returns a JSX element that displays 'Aktualny motyw to: {motyw==='ciemny' ? 'Lord Sith' : 'Jedi'}'.

```
src > components > JS Informacja.js > default
1  import { useContext } from 'react';
2  import { NazwaMotywu } from './App';
3
4  Qodo: Test this function
5  function Informacja() {
6      const motyw = useContext(NazwaMotywu);
7      return(
8          <>
9              <div>Aktualny motyw to: {motyw==='ciemny' ? "Lord Sith" : "Jedi"}</div>
10             </>
11         )
12     }
13     export default Informacja;
```

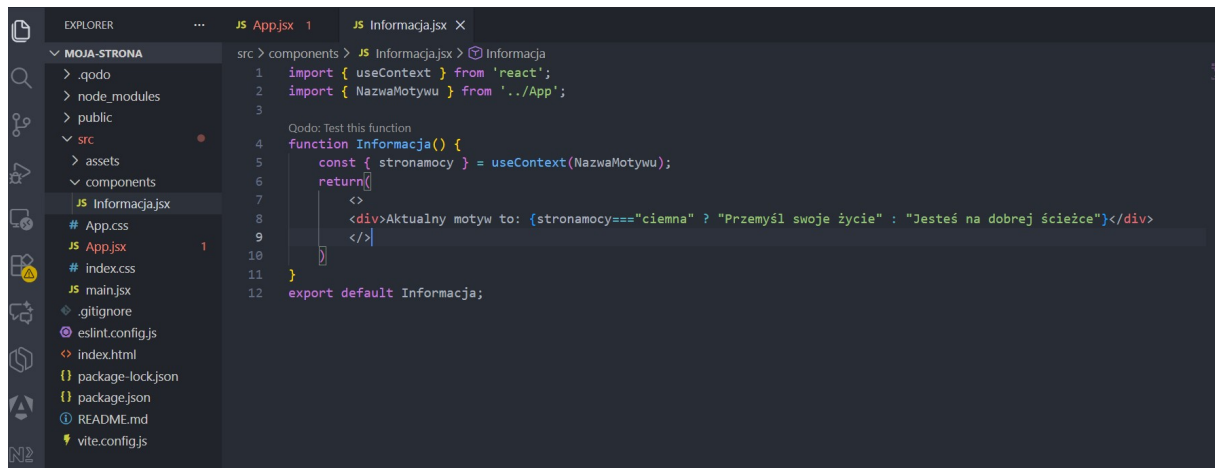
## Zadanie 2 „useContext”

### Utwórz nowy projekt StarWars2

Używamy obiektu a nie „initial value w createContext(‘ciemny’)

The screenshot shows a VS Code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with a 'src' directory containing 'App.js'. The code editor shows the implementation of the 'App' component in 'App.js'. It imports 'React' from 'react' and 'Informacja' from './components/Informacja'. It creates a context 'NazwaMotywu' using 'React.createContext()'. The component function 'App' returns a JSX element that wraps the 'Informacja' component with 'NazwaMotywu.Provider' and sets the 'value' to an object with 'stronamocy' property set to 'ciemna'.

```
src > JS App.js > App
1  import React from 'react';
2  import Informacja from './components/Informacja';
3
4  export const NazwaMotywu = React.createContext();
5
6  Qodo: Test this function
7  function App() {
8      return (
9          <>
10              <NazwaMotywu.Provider value={{stronamocy: "ciemna"}}>
11                  <Informacja/>
12              </NazwaMotywu.Provider>
13          </>
14      )
15  }
16  export default App
17
```



The screenshot shows a VS Code editor with a project named 'MOJA-STRONA'. The Explorer sidebar on the left shows the file structure: `src > components > Informacja`. The code editor displays the `Informacja.js` file with the following code:

```
1 import { useContext } from 'react';
2 import { NazwaMotywu } from '../App';
3
4 Qodo: Test this function
5 function Informacja() {
6   const { stronomocy } = useContext(NazwaMotywu);
7   return(
8     <>
9     <div>Aktualny motyw to: {stronomocy==='ciemna' ? "Przemyśl swoje życie" : "Jesteś na dobrej ścieżce"}</div>
10    </>
11  );
12 }
13 export default Informacja;
```

### Zadanie 3

napisz aplikację w reakcje, która korzysta w całym motywie z określonego zestawu styli:

tło czerwone, zielone, niebieskie

odpowiednio kolor czcionki: biały, pomarańczowy, czarny

Poniżej projekt:

3 komponenty: Header, Main, Footer

`<Header/>`

`<Main/>`

`<Footer/>`

Wszystkie komponenty korzystają z wybranego stylu na podstawie `useContext(styl)`

Główny komponent app posiada pole selekt który zmienia styl I przekazuje dalej przez providera do komponentów

