**Część Praktyczna**

Przygotowanie do zadania

* Wejdź na github: github.com/apraisal.
* Pobierz pliki z folderu **resources**.
* Stwórz swoją apkę reactową ze sposobu z prezentacji albo wejdź na stronę: https://playcode.io/react

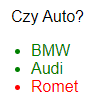
Twoim zadaniem będzie zrobienie strony która wygląda tak:

1. Obraz zawierający tekst, zrzut ekranu, Czcionka, logo

   Opis wygenerowany automatycznieNa stronie musi być jakiś nagłówek.
2. Jeden albo dwa guziki które piszą ile razy się je kliknęło
3. Wyświetlanie listy z (zdj.B) kiedy kliknie się ten guzik przynajmniej 10 razy w miejscu komunikatu.
4. Losowe zdjęcie, tutaj wykorzystałem logo react.
5. Lista powinna zawierać nazwę auta,

Na zielono powinny być auta, a na czerwono wszystko inne

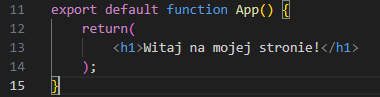
zdj.A



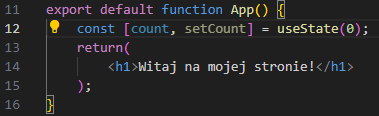
Zdj.B. – przykładowa lista

Możesz spróbować samemu, ale jak nie potrafisz pod spodem masz wskazówki co krok po kroku trzeba zrobić aby osiągnąć taki efekt.

1. W App.js musimy najpierw dodać biblioteki reacta, a także useState: 
2. Zacznijmy od nagłówka w tym celu stworzymy (default) funkcje App która będzie nam zwracać <h1>”nagłówek”</h1> JSX markup.



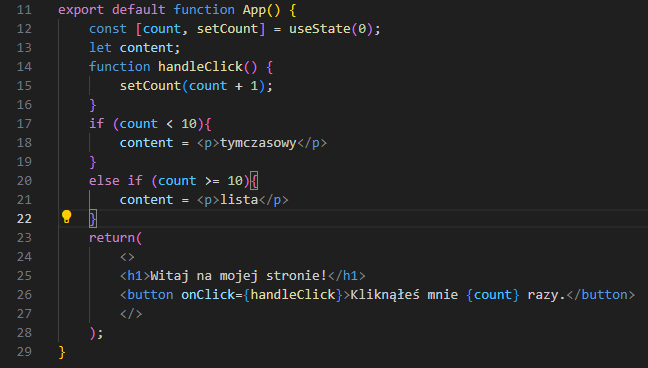
1. Teraz guzik, najpierw stwórzmy zmienna count, możliwość jej zmiany setCount, a także ustawmy ją jako default 0.



1. Następnie dodajmy guzik to funkcji zwracającej i stwórzmy event: click, żeby po kliknięciu guzika zwiększył się count o 1. (pamiętaj o dodaniu pustych znaczników albo div’ów do kodu JSX.)



1. Zajmijmy się teraz stworzeniem warunku na wyświetlanie komunikatu bądź listy. Wykorzystamy do tego konstrukcje **if.** (na razie są tu tylko tymczasowe dane)

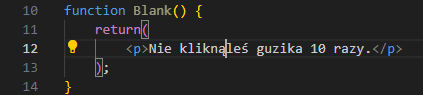


1. Następnie stwórzmy dwie funkcje:

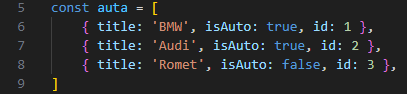
<Blank/> : aby wyświetlać komunikat

<List/> : aby wyświetlać listę

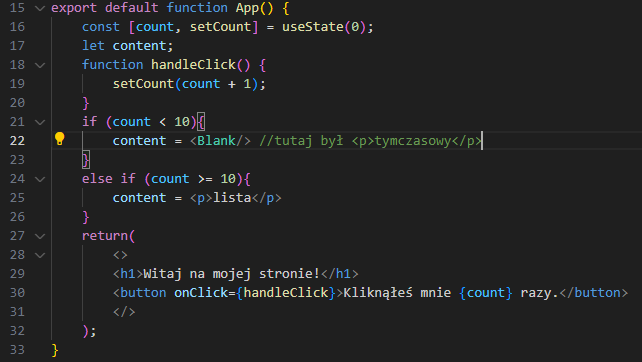
Zacznijmy od <Blank />



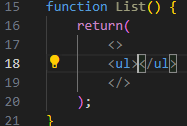
1. Dodajmy teraz listę Auta do app.js (możesz zrobić własną tutaj masz przykładową)



1. Teraz możemy ją zamienić miejscami z tymczasowym contentem.

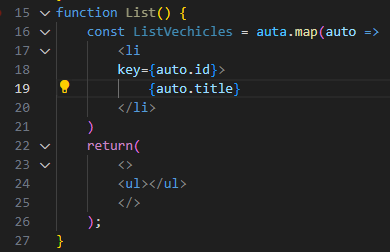


1. Zajmijmy się teraz funkcją mapowania w tym celu stwórzmy nową funkcję <List/>.



1. Teraz użyjmy .map() do zmapowania wcześniej zrobionej przez nas listy.

W tym celu tworzymy zmienna ListVechicles i przypisujemy jej wartość **nazwalisty.map()**, musisz pamiętać o dodaniu także atrybutu key aby odróżnić jeden element listy od innych.

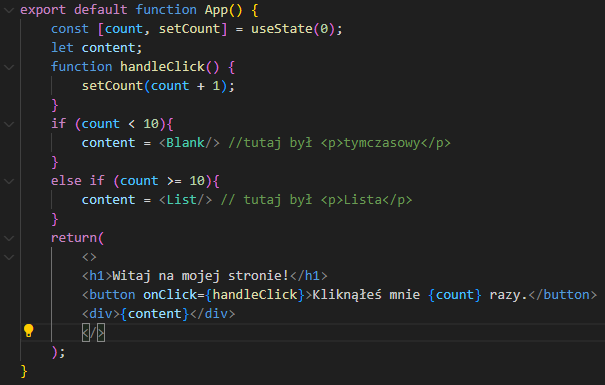


1. Teraz zmieńmy kolor tym elementom dla których isAuto = true.

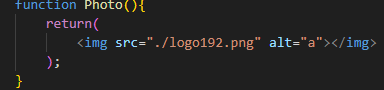
Możemy również stworzyć nagłówek listy za pomocą paragrafu nad nią.

I w funkcji zwracającej w <ul> , wypisz zmienna ListVechicles.



1. Teraz możemy zamienić tymczasowe dane w funkcji podstawowej app. (nie zapomnij dodać do funkcji podstawowej zmiennej content w return.
2. Zajmijmy się teraz ostatnią rzeczą na naszej stronie zdjęciem.

W tym celu stwórzmy nowa funkcję <Photo/> i użyjmy znacznika <img> z HTML i jego atrybutu source (tutaj podajesz source zdjęcia) aby dodać zdjęcie na stronę.



1. Teraz dodajmy funkcje photo do funkcji podstawowej App i całość prezentuje się tak:



1. Jeżeli korzystasz z compilera: playcode.io/react, musisz dodać ten o to kod do index.js:

