

TESTOWANIE



Hello

Kamil Richert

Senior Software Engineer at Atlassian

Rodzaje testów

- **Testy jednostkowe**

Jest, Karma

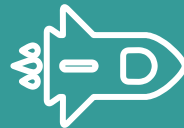
- **Testy integracyjne**

Cypress

- **Testy poziomu end-to-end**

Selenium

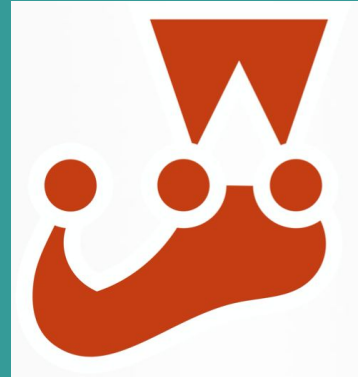
ZADANIE



Zacznijmy od pisania prostych testów w czystym JS.

Wykonaj zadania w folderze: *exercises*.

JEST



JEST

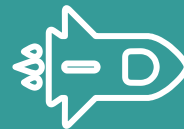
- Framework do testowania rozwijany przez Facebook'a
- Posiada własny test-runner → program do uruchamiania testów
- Posiada własny zbiór asercji (porównań którymi weryfikujemy wyniki testów)
- Daje nam możliwość mierzenia pokrycia testami
- Wbudowane mocki
- Obsługuje wyjątki (te celowe, które chcemy wytestować)
- Można nim testować projekty frontend'owe (używające np Angular'a, Vue czy React'a), czy projekty oparte o Node.js

Przydatne linki:

<https://jestjs.io/docs/en/getting-started> → cała sekcja "Introduction" to podstawy pracy z Jest

<https://create-react-app.dev/docs/running-tests/> → testowanie aplikacji korzystających z create-react-app, gdzie Jest jest wykorzystany jako test runner i główną bibliotekę do testów.

ZADANIE



Wykonaj zadania w folderze: *jest-exercises* oraz *jest-mock-exercises*

REACT TESTING LIBRARY



REACT TESTING LIBRARY

- Nie jest samodzielną biblioteką/frameworkiem do testów (jak jest)
- Daje nam możliwość, podczas testów
 - wyrenderowania komponentu React
 - interakcji z komponentem
 - odczytywania wartości jak z drzewa DOM
- Zbudowane w oparciu o DOM Testing Library

Przydatne linki:

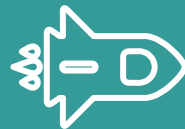
<https://testing-library.com/docs/react-testing-library/intro> → cała sekcja React Testing Library opisuje podstawy pracy z biblioteką

<https://testing-library.com/docs/dom-testing-library/api-queries> → metody budujące wygodne zapytania do DOM

<https://testing-library.com/docs/dom-testing-library/api-events> → odpalanie eventów

<https://create-react-app.dev/docs/running-tests/#react-testing-library> → react-testing-library jest domyślnie w create-react-app

ZADANIE



Wykonaj zadania w folderze: *react-testing-library-exercises*

CYPRESS



CYPRESS

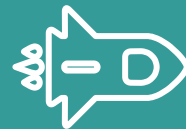
- Kompleksowe rozwiązanie do testów e2e (end-to-end)
- Posiada własny test runner
- Niezależne od aplikacji którą testuje (nie jest istotna technologia użyta w testowanej aplikacji)
- Testy, które “klikają” po stronie - opisujemy te zachowania w JavaScript
- Cypress dba o to aby flow naszych testów podążał za tym jak zmieniają się elementy na stronie - nie musimy ręcznie “czekać” na wykonanie operacji
- Własna biblioteka asercji

Przydatne linki:

<https://docs.cypress.io/guides/getting-started/installing-cypress.html> → cała sekcja Getting Started, pokrywa wszystkie podstawy pracy z Cypress

<https://www.youtube.com/watch?v=LcGHiFnBh3Y> → kompleksowo podstawy Cypress w formie video

ZADANIE



Wykonaj zadania w folderze: *cypress-exercises*



Dzięki

Znajdziecie mnie:

<https://www.linkedin.com/in/kamil-richert/>

<https://github.com/krichert>