



Pystart.pl

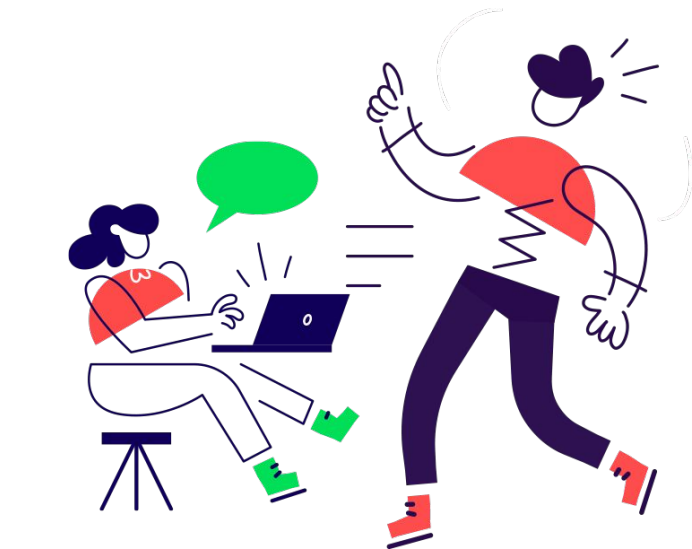
TDD - test driven development

lekcja dwudziesta trzecia

PyStart #23 TDD - test driven development

Dlaczego testujemy?

- Mamy pewność, że nasz program działa.
- Upewniamy się, że nic nie zepsuliśmy.
- Wprowadzamy zmiany dużo szybciej,
nie musimy wszystkiego sprawdzać za każdym razem.



PyStart #23 TDD - test driven development

Co warto testować?

- Logikę naszej aplikacji.
- Kod, który napisaliśmy sami.
- Kod, który modyfikujemy aby upewnić się, że nic nie zepsuliśmy.



PyStart #23 TDD - test driven development

Jak testować?

Instalujemy bibliotekę

```
pip install pytest
```

Uruchamiamy testy

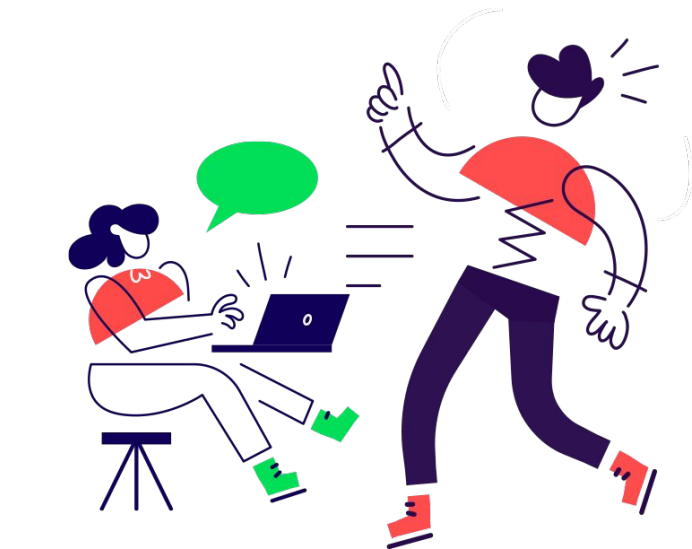
```
python -m pytest program.py
```



PyStart #23 TDD - test driven development

Nic się nie dzieje... brakuje testów

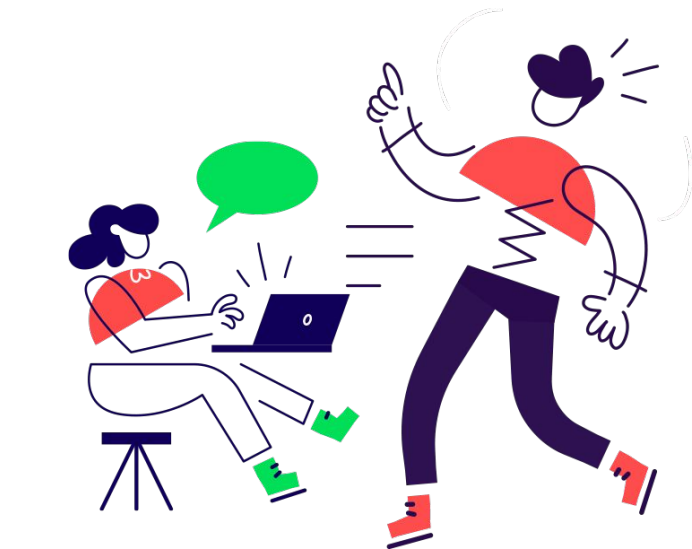
- Test to zwykła funkcja zaczynająca się od **test_**
- Testy zawierają kod napisany w pythonie :)
- Dobry test składa się z trzech części:
 - ◆ Given
 - ◆ When
 - ◆ Then



PyStart #23 TDD - test driven development

Nic się nie dzieje... brakuje testów

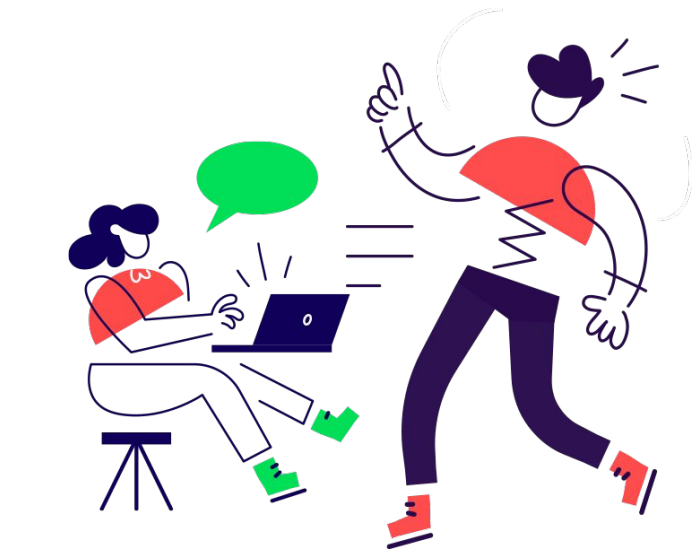
- Test to zwykła funkcja zaczynająca się od **test_**
- Testy zawierają kod napisany w pythonie :)
- Dobry test składa się z trzech części:
 - ◆ Given na dworze jest zimno, ja nie mam czapki
 - ◆ When
 - ◆ Then



PyStart #23 TDD - test driven development

Nic się nie dzieje... brakuje testów

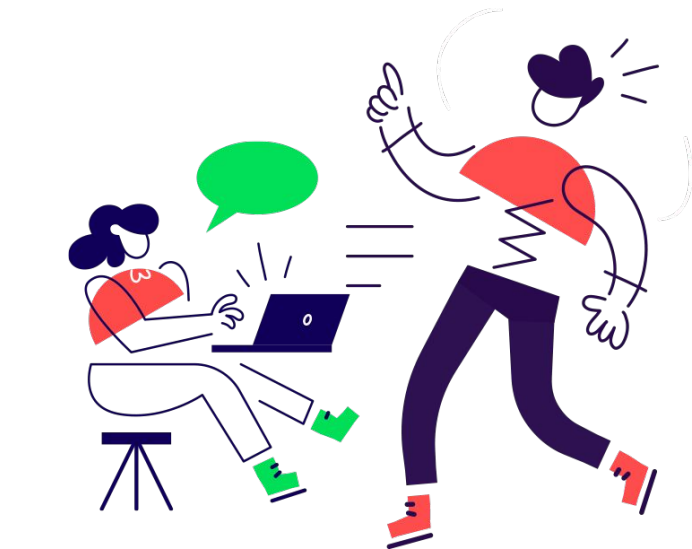
- Test to zwykła funkcja zaczynająca się od **test_**
- Testy zawierają kod napisany w pythonie :)
- Dobry test składa się z trzech części:
 - ◆ **Given** na dworze jest zimno, ja nie mam czapki
 - ◆ **When** wychodzę z domu na dwie godziny
 - ◆ **Then**



PyStart #23 TDD - test driven development

Nic się nie dzieje... brakuje testów

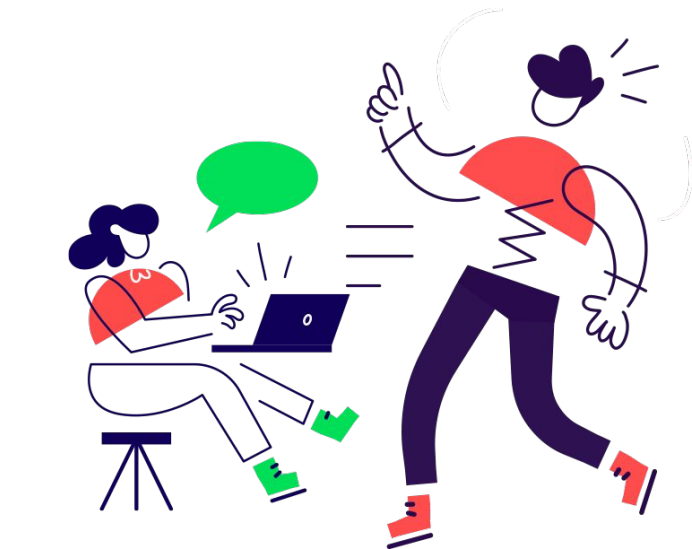
- Test to zwykła funkcja zaczynająca się od **test_**
- Testy zawierają kod napisany w pythonie :)
- Dobry test składa się z trzech części:
 - ◆ **Given** na dworze jest zimno, ja nie mam czapki
 - ◆ **When** wychodzę z domu na dwie godziny
 - ◆ **Then** następnego dnia boli mnie głowa



PyStart #23 TDD - test driven development

Jak wygląda przykładowy test?

```
1 def add_numbers(a: int, b: int) -> int:
2     return a + b
3
4
5 def test_add_numbers():
6     a = 2
7     b = 3
8
9     value = add_numbers(a, b)
10
11     assert value == 5
12
13
```



PyStart #23 TDD - test driven development

Gdy wszystko jest ok

→ Tak wygląda szczęście ;-)

```
(venv) D:\Trainings\Pystart>python -m pytest trening.py
===== test session starts =====
platform win32 -- Python 3.8.5, pytest-6.2.3, py-1.10.0, pluggy-0.13.1
rootdir: D:\Trainings\Pystart
collected 1 item

trening.py . [100%]

===== 1 passed in 0.01s =====
(venv) D:\Trainings\Pystart>
```



PyStart #23 TDD - test driven development

Gdy wszystko jest ok



→ Tak wygląda błąd ;-)

```
===== FAILURES =====
test_add_numbers

def test_add_numbers():
    a = 2
    b = 3

    value = add_numbers(a, b)

>     assert value == 8
E     assert 5 == 8

trening.py:11: AssertionError
===== short test summary info =====
FAILED trening.py::test_add_numbers - assert 5 == 8
===== 1 failed in 0.06s =====
```

PyStart #23 TDD - test driven development

Zróbmy jeszcze raz zaczynając od testów...

- Przygotuj funkcję, która zliczy ilość znaków w tekście zawierających się wewnątrz nawiasów okrągłych. Nawiasy mogą występować w tekście wielokrotnie, nigdy nie będą się w sobie zawierać.

```
1 def count_letters(text, start='(', end=')'):  
2     pass  
3  
4  
5     count_letters('ała) ma (kota)')  
6     # zwróci 3 + 4  
7  
8     count_letters('<> kod <103>', '<', '>')  
9     # zwróci 3  
10  
11     count_letters('abrakadabra')  
12     #zwróci 0
```



PyStart #23 Wchodzimy w funkcje

Zadania dla nabrania wprawy

!!! PAMIĘTAJ O TYM BY ROZPOCZĄĆ OD TESTÓW !!!

23.1

1. Przygotuj funkcję, która usunie wszystkie samogłoski z przekazanego do funkcji napisu.
2. Przygotuj funkcję, którą nazwij **play_game**. Jej argumenty to **player_choice** **computer_choice**. Do wyboru “kamień”, “papier”, “nożyce”. Jeśli wygra gracz funkcja powinna zwrócić 1, jeśli komputer funkcja zwraca 2, jeśli remis 0.

