

## Przygotowanie środowiska PyCharm do zajęć:

- Testy jako obywatel pierwszej kategorii z Test-Driven Development
- Tester jako developer narzędzi z pomocą Python-a – warsztaty

Pracujemy w programie PyCharm w środowisku VDI Wyższej Szkoły Bankowej:

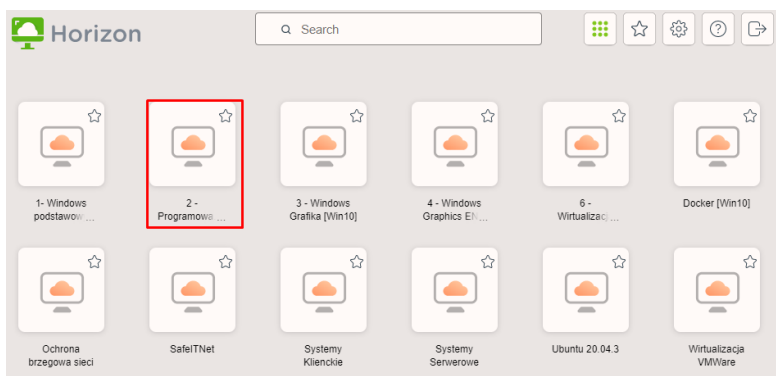
(można pracować w dowolnym lokalnym środowisku, trzeba je wtedy poprawnie skonfigurować – porady, jak skonfigurować własne środowisko znajdują się na ostatniej stronie)

1. Logujemy się na: <https://vlabs.wsb.wroclaw.pl/portal/webclient/#/home>

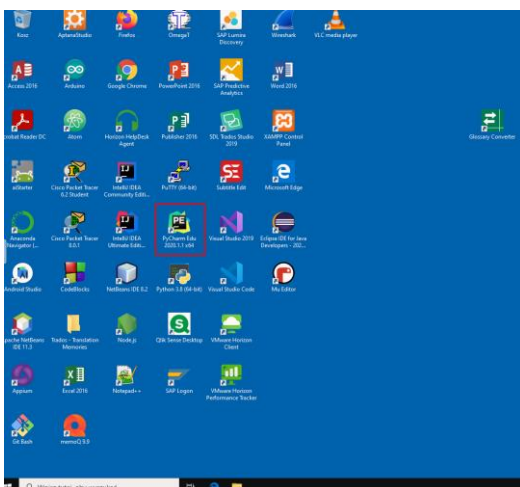
Użytkownik: vdi-student

Hasło: Pa\$\$w0rd

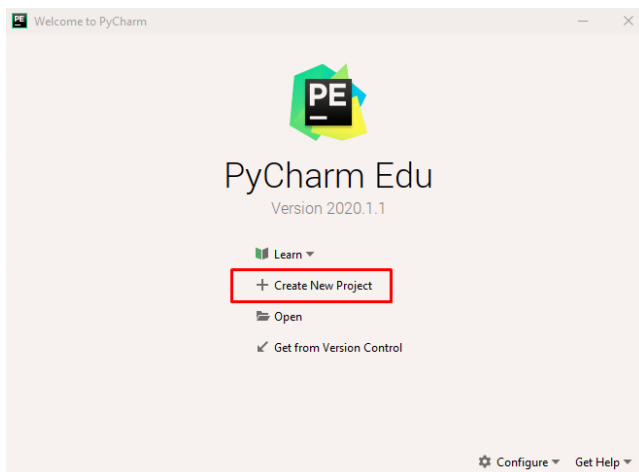
2. Wybieramy obraz **2-Programowanie**



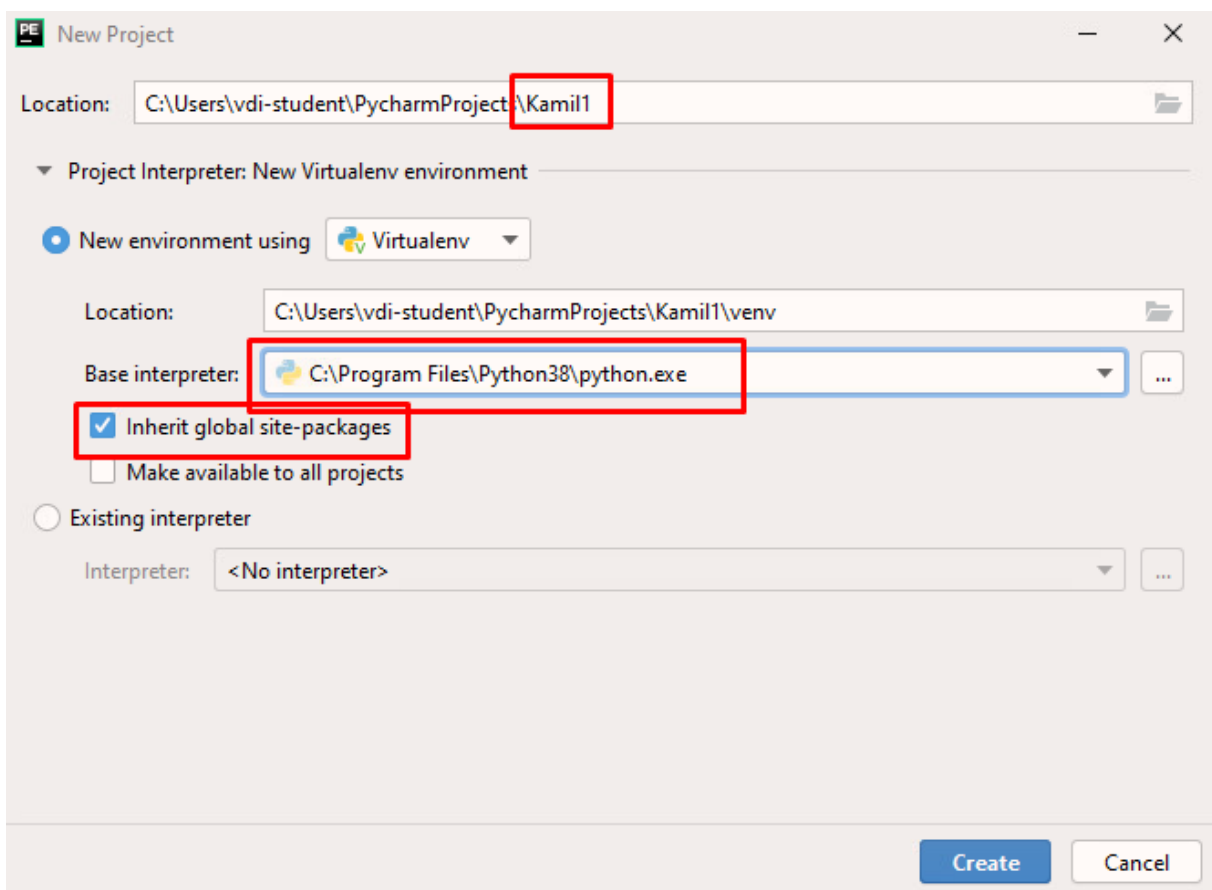
3. Uruchamiamy PyCharm



4. Tworzymy nowy projekt:

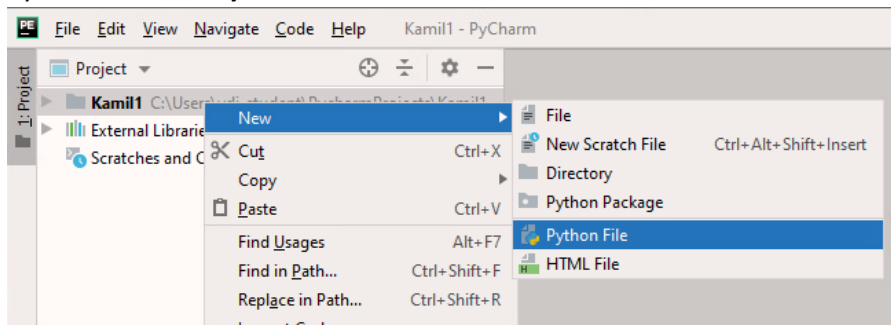


5. Wybieramy dowolną nazwę (u mnie Kamil1), ustawiamy **Base interpreter** Python38 oraz zaznaczamy **Inherit global site-packages**



6. Dla zajęć Tester jako developer narzędzi z pomocą Python-a tworzymy jeden plik „Python File”.

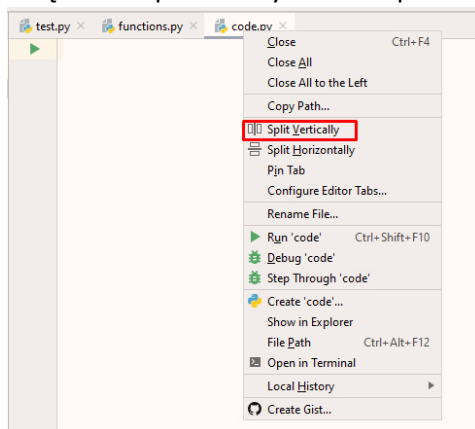
Klikamy prawym przyciskiem myszy na nazwie projektu (u mnie **Kamil1**) i tworzymy nowy plik Python – **New -> Python File**



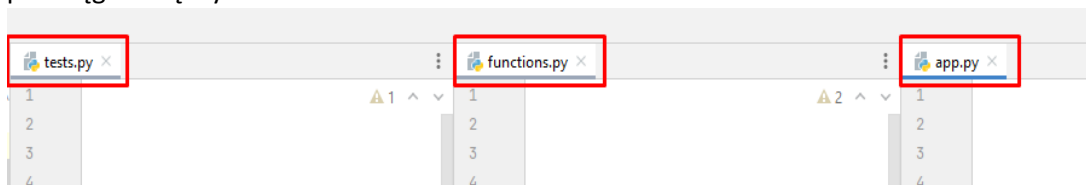
Gotowe 😊

7. Dla zajęć Testy jako obywatel pierwszej kategorii z Test-Driven Development tworzymy 3 pliki w naszym projekcie: app, functions, tests (instrukcja w punkcie 6)

8. Klikamy prawym przyciskiem myszy na jedną z zakładek i wybieramy **Split Vertically**  
Dzięki temu podzielimy widok w pionie



9. Powtarzamy operację z 2gą zakładką, aby mieć każdą w osobnej kolumnie. Zakładki można przeciągać między kolumnami



Gotowe 😊

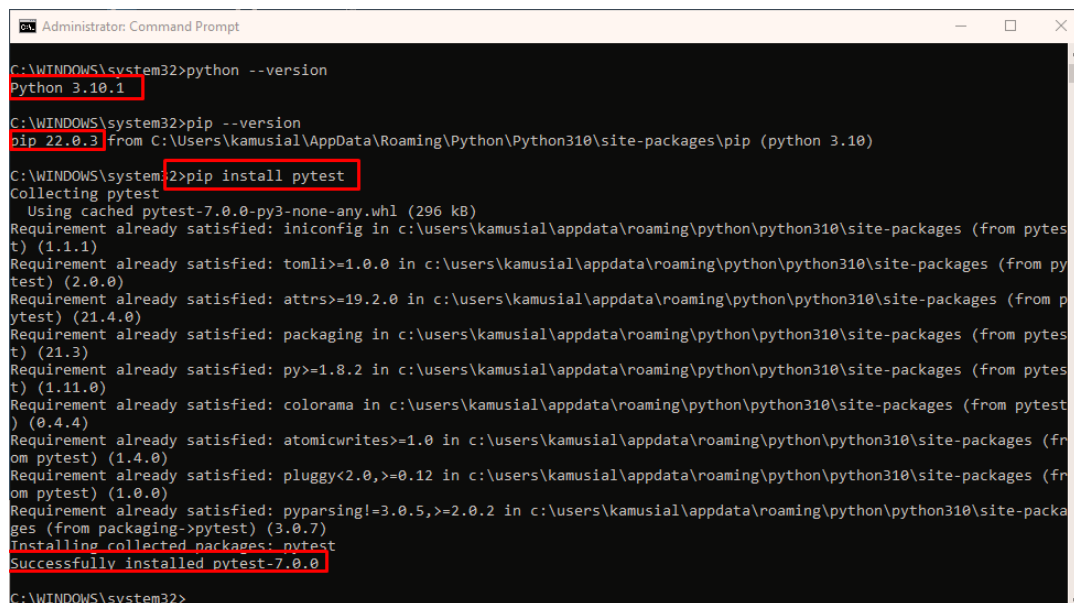
## Własne środowisko:

Działając na obrazie VDI mamy zainstalowane wszystko, co potrzebne.

Działając we własnym lokalnym środowisku należy pobrać i zainstalować odpowiednie narzędzia:

1. Python: <https://www.python.org/downloads/>
2. PyCharm lub Visual Studio Code lub PyScripter lub Mu lub inne
3. Biblioteka **pytest** – globalnie w systemie lub w danym projekcie

3a. Aby zainstalować bibliotekę **pytest** w systemie, wchodzimy do **cmd** jako administrator. Upewniamy się, że jest zainstalowany Python oraz pip i instalujemy bibliotekę komendą:  
**pip install pytest**



```
C:\WINDOWS\system32>python --version
Python 3.10.1

C:\WINDOWS\system32>pip --version
pip 22.0.3 from C:\Users\kamusial\AppData\Roaming\Python\Python310\site-packages\pip (python 3.10)

C:\WINDOWS\system32>pip install pytest
Collecting pytest
  Using cached pytest-7.0.0-py3-none-any.whl (296 kB)
Requirement already satisfied: iniconfig in c:\users\kamusial\appdata\roaming\python\python310\site-packages (from pytest) (1.1.1)
Requirement already satisfied: tomli>=1.0.0 in c:\users\kamusial\appdata\roaming\python\python310\site-packages (from pytest) (2.0.0)
Requirement already satisfied: attrs>=19.2.0 in c:\users\kamusial\appdata\roaming\python\python310\site-packages (from pytest) (21.4.0)
Requirement already satisfied: packaging in c:\users\kamusial\appdata\roaming\python\python310\site-packages (from pytest) (21.3)
Requirement already satisfied: py>=1.8.2 in c:\users\kamusial\appdata\roaming\python\python310\site-packages (from pytest) (1.11.0)
Requirement already satisfied: colorama in c:\users\kamusial\appdata\roaming\python\python310\site-packages (from pytest) (0.4.4)
Requirement already satisfied: atomicwrites>=1.0 in c:\users\kamusial\appdata\roaming\python\python310\site-packages (from pytest) (1.4.0)
Requirement already satisfied: pluggy<2.0,>=0.12 in c:\users\kamusial\appdata\roaming\python\python310\site-packages (from pytest) (1.0.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in c:\users\kamusial\appdata\roaming\python\python310\site-packages (from packaging->pytest) (3.0.7)
Installing collected packages: pytest
Successfully installed pytest-7.0.0

C:\WINDOWS\system32>
```

3b. Aby zainstalować bibliotekę wewnątrz danego projektu wchodzimy w odpowiednie ustawienia (zależnie od programu, w przykładzie PyCharm)

