



SMARTFACTOR

BootCamp: zajęcia 1

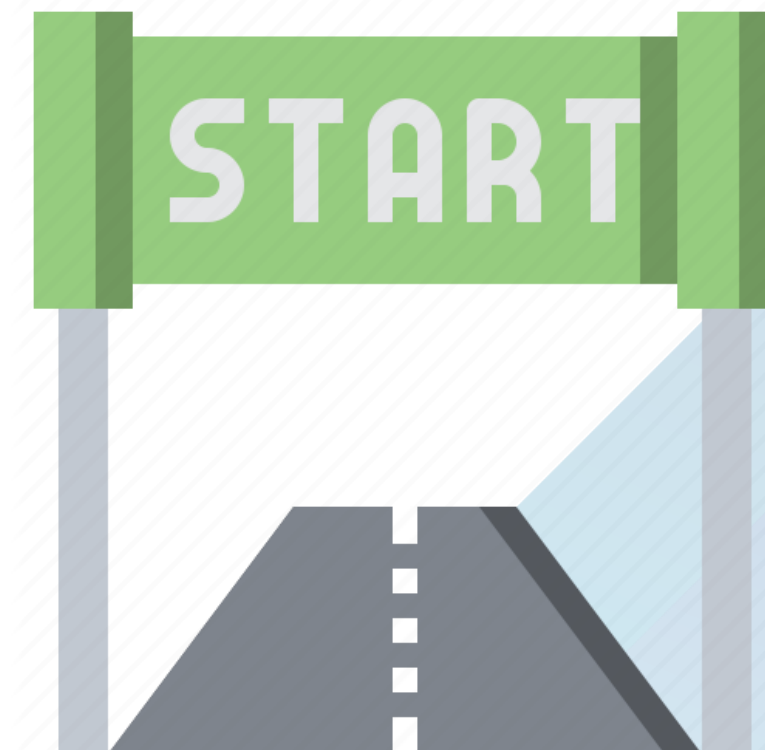
„Nie zawsze potrafię przewidywać,
ale potrafię **kłaść podwaliny**.
Bo **przyszłość** jest czymś,
co się **buduje**”

Antoine de Saint-Exupery

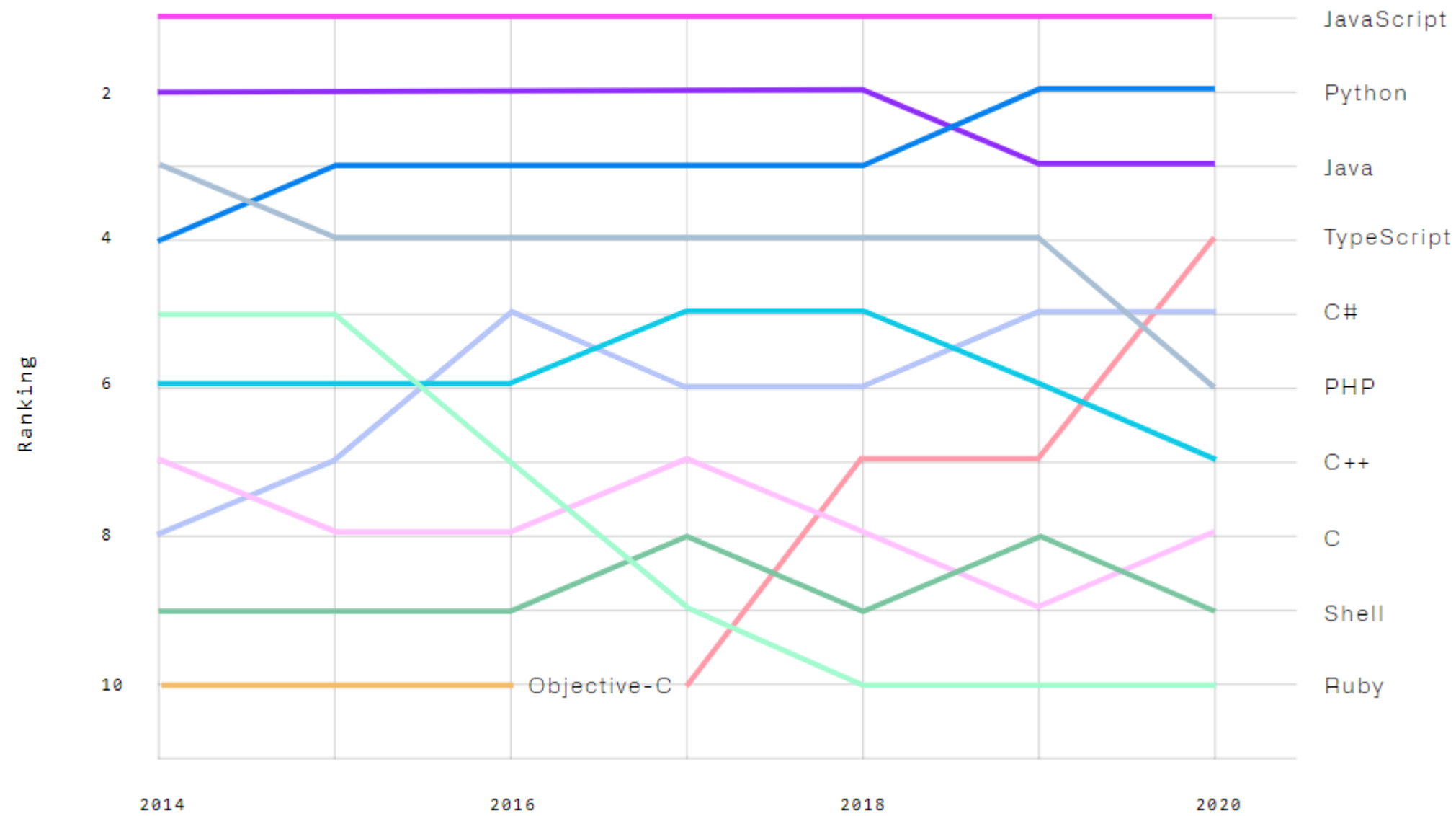


Spis treści

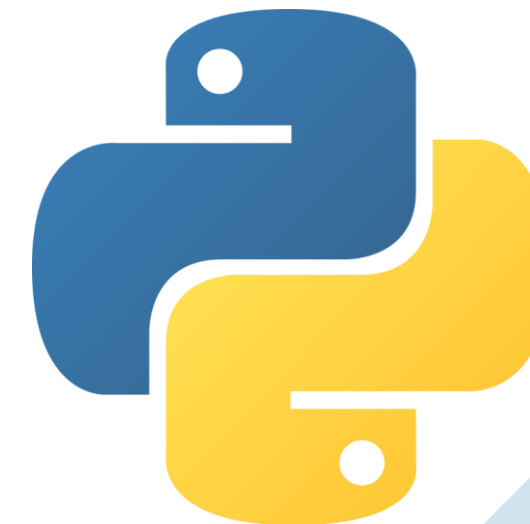
1. Dlaczego warto się uczyć Pythona?
2. Do jakich celów używany jest Python?
3. Jak będziemy pracować?
4. Instalacja oprogramowania
5. Tworzenie wirtualnego środowiska pracy
6. Instalacja IDE
7. Typy zmiennych w Pythonie, podstawowe funkcje języka Python
8. Zadania do samodzielnego rozwiązania
9. Przydatne linki



Dlaczego warto się uczyć Python'a?



Źródło: The 2020 State of the Octoverse



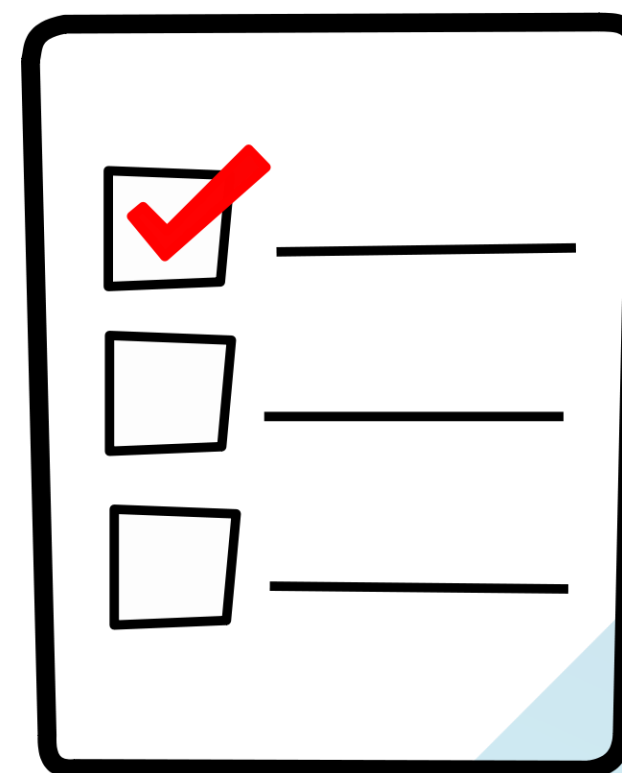
Do jakich celów używany jest Python?

- Data Science 🗨 pandas, geopandas, matplotlib 🗨
- Uczenie maszynowe 🗨 sciki-learn, keras, tensorflow 🗨
- Widzenie komputerowe 🗨 OpenCV
- Web Development – Django
- Bazy Danych
- Obliczenia macierzowe – numpy
- Tworzenie gier
- Aplikacje okienkowe – TkInter
- ...



Jak będziemy pracować?

1. Warsztaty on-line (2-3 godz.)
2. Przekazanie uczestnikom zadań do rozwiązania i materiałów z zajęć
3. Praca własna z możliwością konsultacji na czacie lub głosowo do godz. 17:00
4. Przesłanie przez uczestników wyników pracy
5. Sprawdzenie zadań i informacja zwrotna



Gdzie będziemy pracować?





Instalacja oprogramowania

Conda

latest

Search docs

Conda

Conda-build

Miniconda

Windows installers

MacOSX installers

Linux installers

Installing

Other resources

Help and support

Contributing

Conda license

Docs » Miniconda

Edit on GitHub

Miniconda

Miniconda is a free minimal installer for conda. It is a small, bootstrap version of Anaconda that includes only conda, Python, the packages they depend on, and a small number of other useful packages, including pip, zlib and a few others. Use the `conda install command` to install 720+ additional conda packages from the Anaconda repository.

See if Miniconda is right for you.

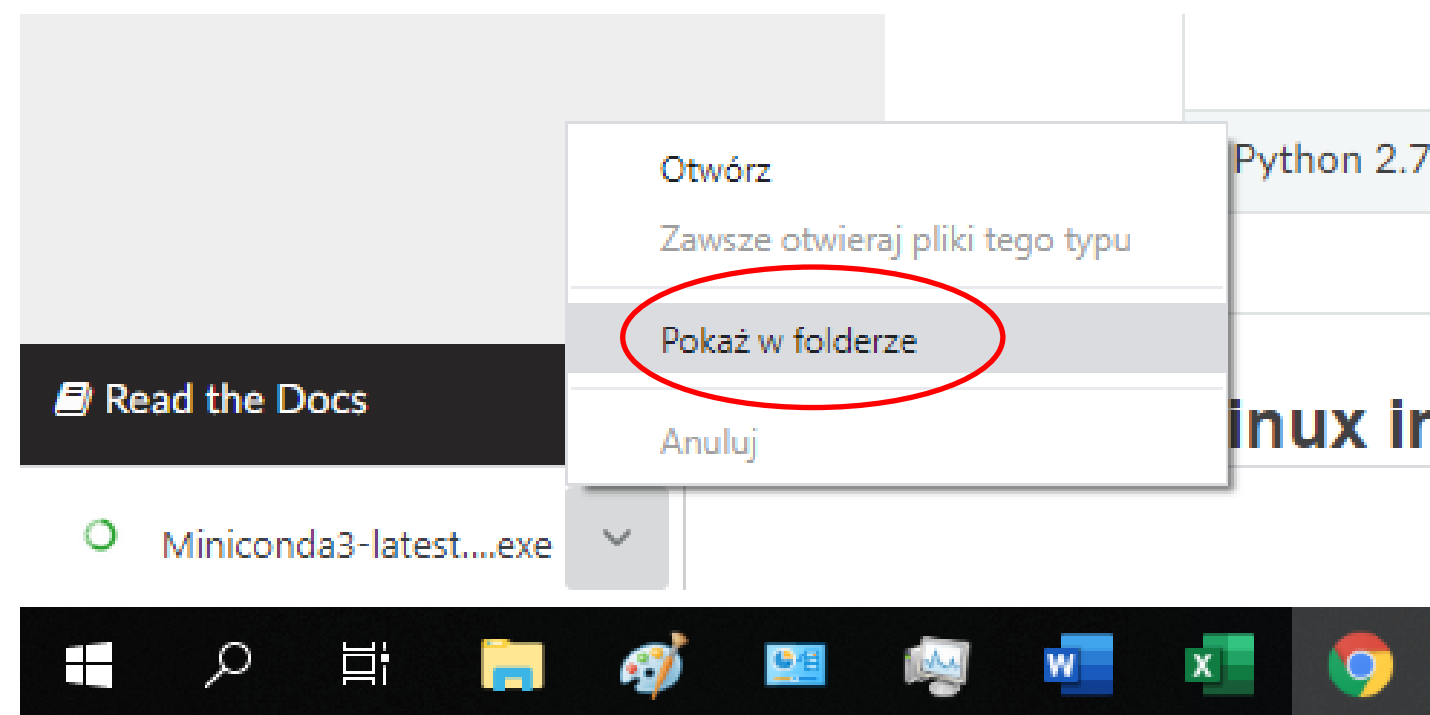
Windows installers

Windows			
Python version	Name	Size	SHA256 hash
Python 3.8	Miniconda3 Windows 64-bit	57.0 MiB	4fa22bba0497babb5b6608cb8843545372a99f5331c8120099ae1d803f627c61
	Miniconda3 Windows 32-bit	54.2 MiB	9c2ef76bae97246c85c206733ca30fd1feb8a4b3f90a2a511fea681ce7ebc661
Python 2.7	Miniconda2 Windows 64-bit	54.1 MiB	6973025404832944e074bf02bda8c4594980eed4707bb51baa8fbd4b326c
	Miniconda2 Windows 32-bit	47.7 MiB	c8049d26f8b6b954b57bcd4e99ad72d1ffa13f4a6b218e64e641504437b2617b

[Miniconda — Conda documentation](#) (link do pobierania instalatorów)

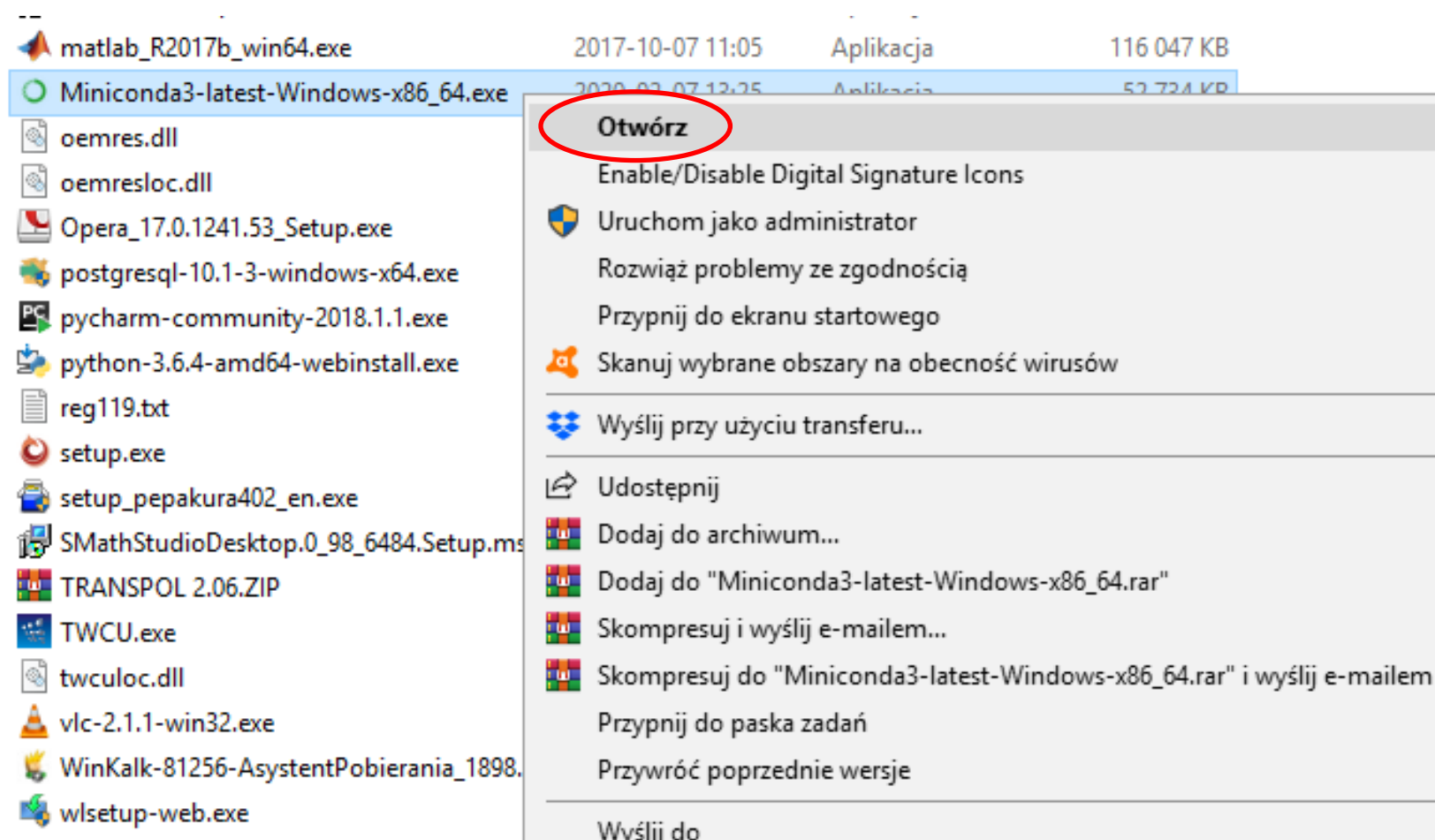
Pobierz plik instalacyjny dla Pythona w wersji 3.8 dla swojego systemu operacyjnego (zazwyczaj Windows 64-bit)

Instalacja oprogramowania

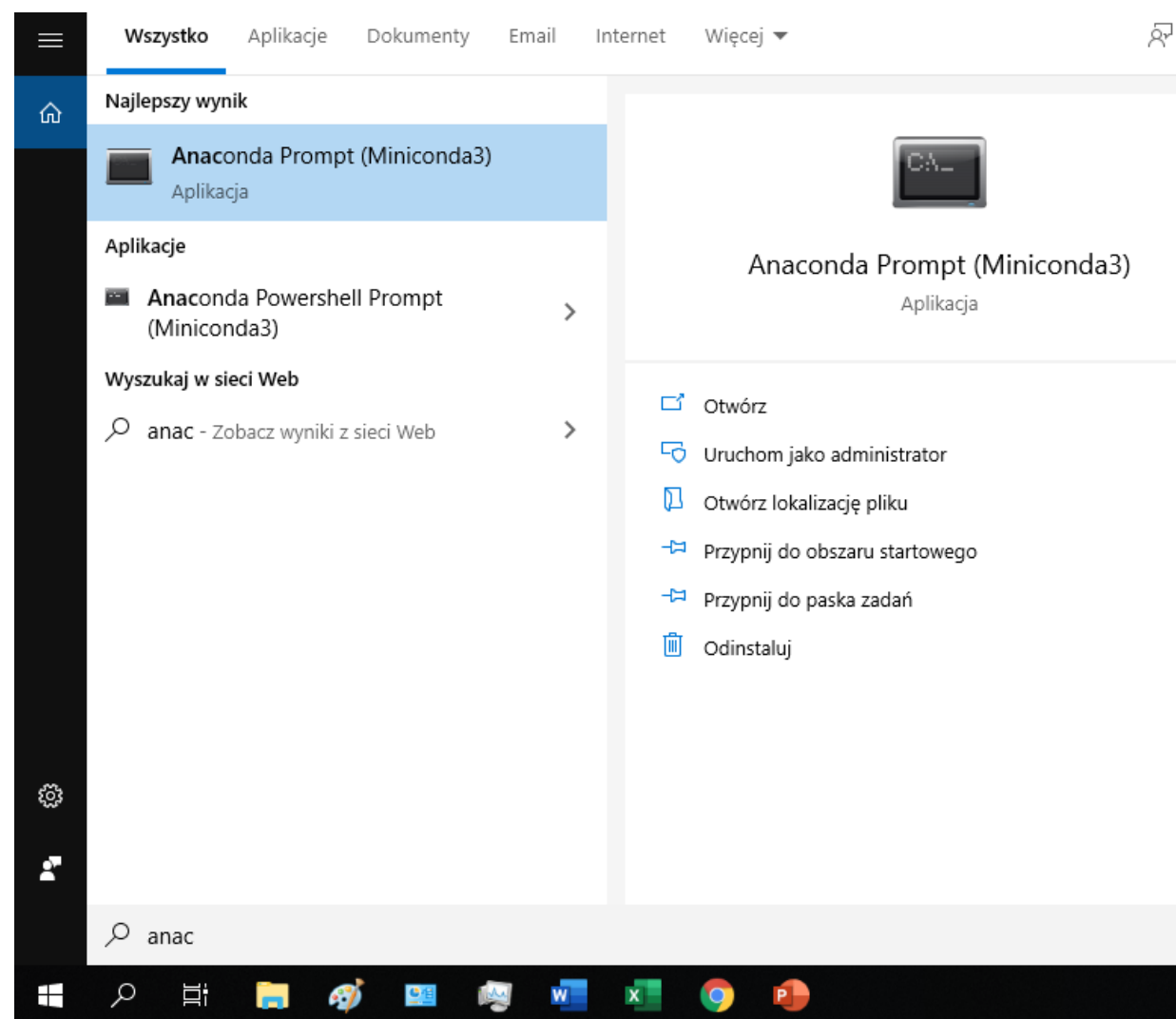


Naciśnij prawym przyciskiem myszy i zainstaluj program (wszystkie opcje pozostawiamy domyślnie)

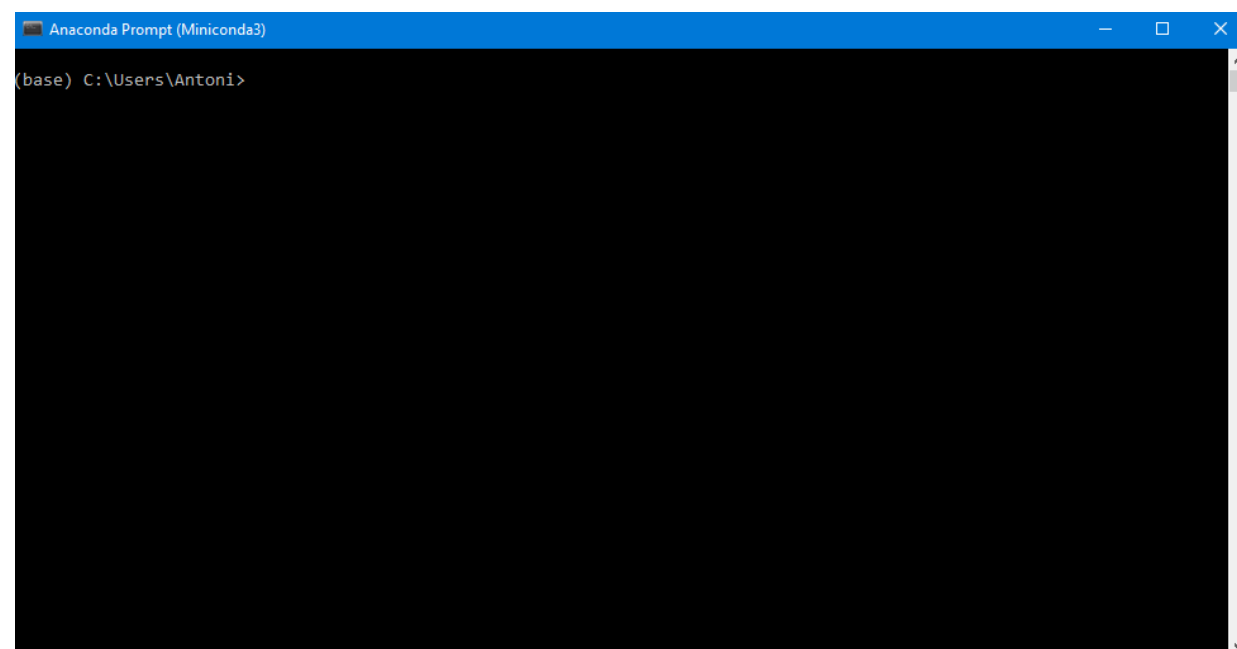
Po pobraniu przejdź do folderu, gdzie został zapisany plik instalacyjny



Instalacja oprogramowania



Po ukończeniu instalacji wyszukaj w menu Start aplikację Anaconda Prompt (Miniconda 3)



Po uruchomieniu aplikacji powinno otworzyć się okno konsoli. Gotowe, możemy zaczynać!


Co jeśli nie zadziała?

- ! Pamiętaj, że zawsze możesz skorzystać z interpretera kodu Pythona on-line (będzie to nasza opcja awaryjna). Jeśli nie uda Ci się zainstalować Minicondy przed pierwszymi zajęciami to nic straconego, możesz tworzyć i uruchamiać skrypty pod tym linkiem (konieczne jest założenie konta):
[Repl.it - The collaborative browser based IDE](https://repl.it)
- ! Pamiętaj, że zawsze możesz poprosić o konsultacje i pomoc, wystarczy napisać i umówić się na rozmowę 😊




Tworzenie wirtualnego środowiska pracy

Wszystko co powinniśmy wiedzieć:

[Managing environments — conda 4.9.2.post24+e37cf84a documentation](#) 

Managing environments

- 
- Creating an environment with commands
 - Creating an environment from an environment.yml file
 - Specifying a location for an environment
 - Updating an environment
 - Cloning an environment
 - Building identical conda environments
 - Activating an environment
 - Deactivating an environment
 - Determining your current environment
 - **Viewing a list of your environments**
 - Viewing a list of the packages in an environment
 - Using pip in an environment
 - Setting environment variables
 - Saving environment variables
 - Sharing an environment
 - Restoring an environment
 - Removing an environment

Miejsce na notatki

A large, empty rectangular area with a dashed blue border, intended for taking notes.

Instalacja IDE (Integrated Development Environment)



```
conda install -c anaconda spyder
```



```
conda install -c conda-forge jupyterlab
```



SMARTFACTOR

Instalacja IDE: interfejs Spydera

Spyder (Python 3.7)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\antek

C:\Users\antek\spyder-py3\temp.py

temp.py

1

Nowy plik

Uruchom skrypt

Usuń zmienne

Obszar do tworzenia skryptów

Okno podglądu plików, zmiennych, pomocy i wykresów

Console 1/A

Python 3.7.9 (default, Aug 31 2020, 17:10:11) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.18.1 -- An enhanced Interactive Python.

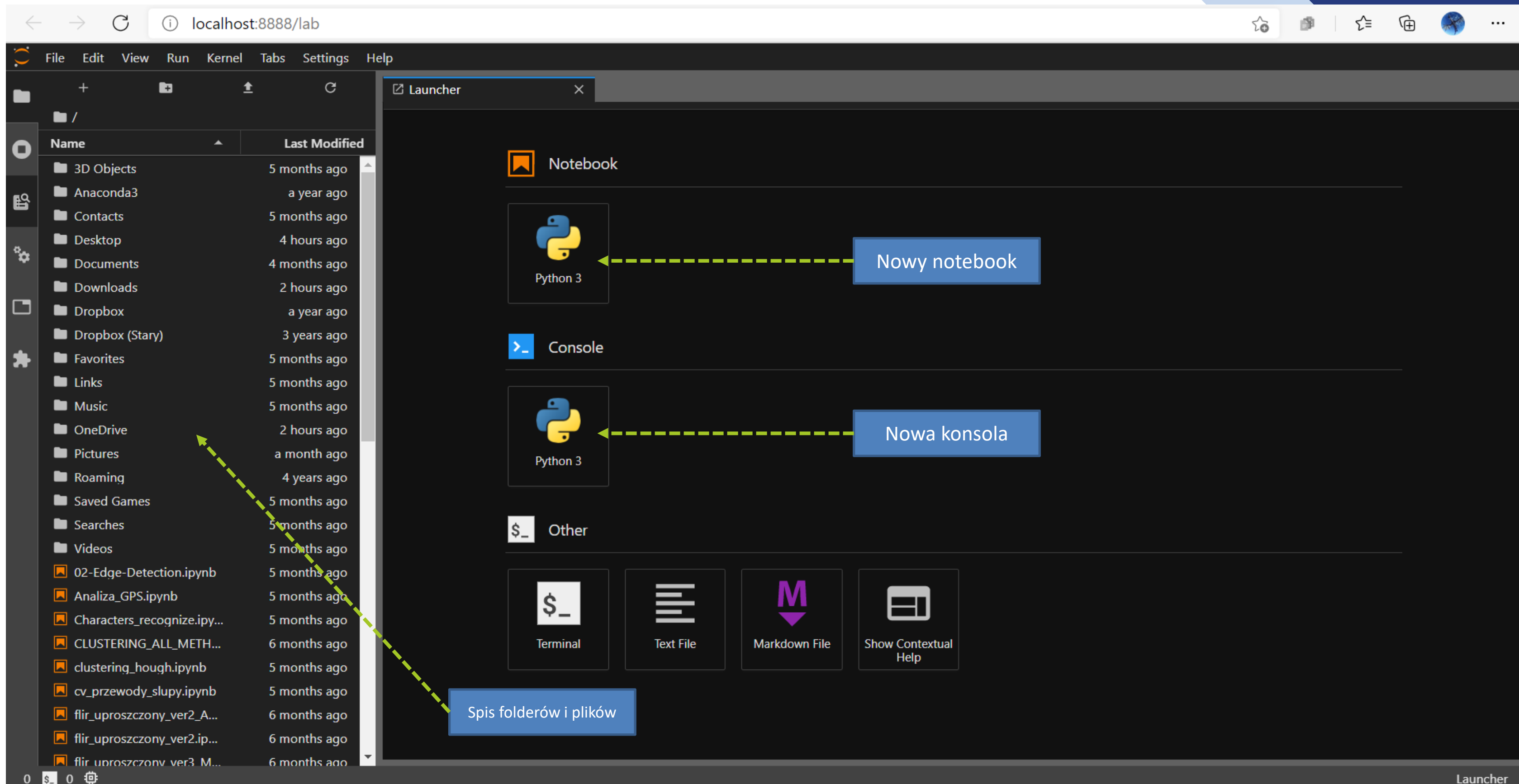
In [1]:

Okno konsoli

IPython console History

LSP Python: ready conda: myvenv (Python 3.7.9) Line 1, Col 1 UTF-8-GUESSED CRLF RW Mem 75%

Instalacja IDE: interfejs Jupyter Lab

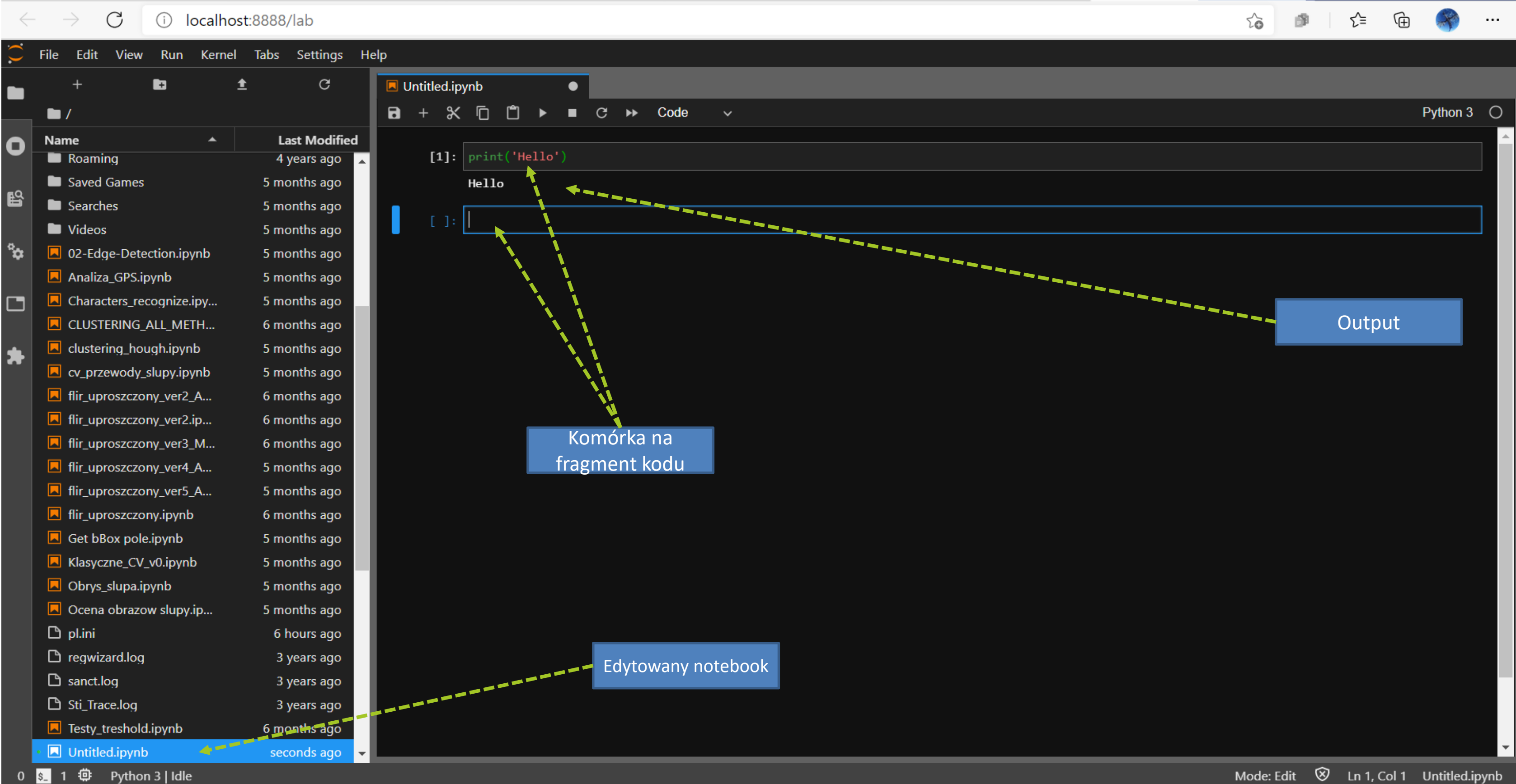


The screenshot displays the Jupyter Lab web interface in a browser window at `localhost:8888/lab`. The interface is divided into three main sections:

- File Browser (Left):** A sidebar showing a file system tree. It includes standard system folders like Desktop, Documents, and Downloads, as well as Jupyter Lab notebooks. A dashed green arrow points from the annotation "Spis folderów i plików" to this sidebar.
- Launcher (Center):** A central panel with a dark theme. It contains three sections:
 - Notebook:** Features a Python 3 icon. A dashed green arrow points from the annotation "Nowy notebook" to this icon.
 - Console:** Features another Python 3 icon. A dashed green arrow points from the annotation "Nowa konsola" to this icon.
 - Other:** Contains icons for Terminal, Text File, Markdown File, and Show Contextual Help.
- Top Menu:** A horizontal bar with menus for File, Edit, View, Run, Kernel, Tabs, Settings, and Help.

At the bottom right of the interface, the word "Launcher" is visible.

Instalacja IDE: interfejs Jupyter Lab



The screenshot shows the Jupyter Lab web interface in a browser at `localhost:8888/lab`. The interface is divided into three main panels: a file browser on the left, a code editor in the center, and a console/output area on the right.

File Browser (Left Panel): Displays a list of files and folders. The 'Name' and 'Last Modified' columns are visible. The file `Untitled.ipynb` is selected at the bottom of the list, marked as 'seconds ago'.

Code Editor (Center Panel): Shows a notebook with two code cells. The first cell contains the code `[1]: print('Hello')` and has executed, displaying the output `Hello`. The second cell is empty and ready for input, indicated by a blue cursor. A dashed yellow arrow points from the text 'Komórka na fragment kodu' to the first code cell.

Output (Right Panel): A blue box labeled 'Output' is shown, with a dashed yellow arrow pointing from the text 'Output' to it.

Annotations:

- A dashed yellow arrow points from the text 'Edytowany notebook' to the `Untitled.ipynb` file in the file browser.
- A dashed yellow arrow points from the text 'Komórka na fragment kodu' to the first code cell in the editor.
- A dashed yellow arrow points from the text 'Output' to the output box on the right.

Bottom Status Bar: Displays '0 \$ 1 Python 3 | Idle', 'Mode: Edit', 'Ln 1, Col 1', and 'Untitled.ipynb'.

Miejsce na notatki

A large, empty rectangular area with a dashed blue border, intended for taking notes.

Zmienne w Pythonie, podstawowe funkcje języka Python

- Co to jest zmienna?
 - Funkcja `id()`
 - Jakie mamy typy zmiennych?
 - Jak definiujemy poszczególne typy zmiennych w Pythonie?
 - Podstawowe operacje na zmiennych
 - Metody łańcuchów znaków
 - Metody list
 - Operacje na zbiorach
 - Odwoływanie się do elementów list i słowników
 - Konwertowanie zmiennych, funkcje: `int()`, `float()`, `list()`, `set()`
 - Podstawowe funkcje języka Python: `print()`, `type()`, `input()`, `min()`, `max()`
-
- Notatki w Jupyter Lab – dobry sposób nauki

Miejsce na notatki

A large, empty rectangular area with a dashed blue border, intended for taking notes.

Miejsce na notatki

A large, empty rectangular area with a dashed blue border, intended for taking notes.

Zadania do samodzielnego rozwiązania

Zadanie 1.

Stwórz po jednej zmiennej o każdym typie, który poznałeś. Pokaż ich wartości w oknie komend za pomocą funkcji *print* oraz ich typ za pomocą funkcji *type*.

Zadanie 2.

Stwórz dwie zmienne, które są liczbami. Wykonaj na nich działania: dodawanie, odejmowanie, mnożenie, dzielenie, potęgowanie.

Zadanie 3.

Za pomocą funkcji *input* pobierz od użytkownika jedną zmienną i poinformuj go komunikatem jaki jest typ tej zmiennej.

Zadanie 4.

Stwórz zmienną o typie *int*. Zmień jej typ na *float* za pomocą funkcji *float* a następnie zmień z powrotem typ na *int* za pomocą funkcji *int*.

Zadanie 5.

Stwórz dwie zmienne o typie *str* – swoje imię i nazwisko. Wyświetl je w oknie komend jako jeden ciąg znaków.

Zadanie 6.

Stwórz zmienną o typie *list* zawierającą liczby całkowite. Przetestuj na niej działanie metod *append*, *index* oraz *count*.

Zadania do samodzielnego rozwiązania

Zadanie 7.

Stwórz listę zawierającą 15 liczb. Za pomocą funkcji *max* wybierz największą wartość listy. Wyświetl ją na konsoli.

Zadanie 8.

Stwórz listę zawierającą 15 liczb. Za pomocą funkcji *min* wybierz najmniejszą wartość listy. Wyświetl ją na konsoli.

Zadanie 9.

Stwórz dwie zmienne o typie *set*. Niech zawierają po dziesięć dowolnych liczb z przedziału od zera do piętnastu. Wykonaj kilka operacji na zbiorach i pokaż ich wyniki.

Zadanie 10.

Dane są zbiory $A = \{-5, -3, -1, 1, 3, 5\}$ oraz $B = \{1, 2, 3, 4, 5\}$. Wyznacz zbiory $A \cup B$, $A \cap B$, $A - B$, $B - A$.

Zadanie 11.

Poproś użytkownika o jego imię i nazwisko, za pomocą metody *format* pokaż na konsoli tekst powitalny: Witaj /imię nazwisko/, miło Cię widzieć!

Zadanie 12.

Stwórz słownik zawierający jako klucze kilka angielskich słów, a jako wartości – ich polskie tłumaczenia. Odwołaj się do wybranego klucza i wyświetl na konsoli jego wartość.

Zadania do samodzielnego rozwiązania

Zadanie 13.

Zdefiniuj listę o sześciu elementach. Napisz skrypt, który prosi użytkownika o uzupełnienie listy jeszcze trzema elementami. Po każdorazowym zadeklarowaniu przez użytkownika dopisywanego elementu pokaż aktualną listę.

Zadanie 14.

Przechwyć od użytkownika dowolną liczbę całkowitą. Zmień jej typ na liczbę rzeczywistą i wyświetl nową liczbę na konsoli. W następnej linii wyświetl jej aktualny typ.

Zadanie 15.

Użytkownik podaje długości podstaw oraz wysokość trapezu. Napisz skrypt, który zwraca informację o polu trapezu.

Zadanie 16.

Użytkownik podaje rok swojego urodzenia. Napisz skrypt, który zwraca informację o tym, czy użytkownik jest pełnoletni

Zadanie 17.

Przypomnij sobie proces tworzenia nowego środowiska w Minicondzie oraz „pobaw się” z interfejsem Jupyter Lab, spróbuj stworzyć nowy notebook i uruchomić kilka linii kodu. Jako rozwiązanie zadania wyślij plik stworzonego notebooka (rozszerzenie .ipynb)

PAMIĘTAJ O MOŻLIWOŚCI KONSULTACJI! JEŚLI CHCESZ ZADAĆ PYTANIE ZRÓB TO ŚMIAŁO, PRACA WŁASNA JEST BARDZO WAŻNA ALE PYTAJĄC MOŻESZ DOWIEDZIEĆ SIĘ SZYBKO NOWYCH I PRZYDATNYCH RZECZY ;)

Przydatne linki

- [1] [Project Jupyter | Home](#)
- [2] [Spyder \(oprogramowanie\) – Wikipedia, wolna encyklopedia](#)
- [3] [Repl.it - The collaborative browser based IDE](#)
- [4] [The State of the Octoverse | The State of the Octoverse explores a year of change with new deep dives into developer productivity, security, and how we build communities on GitHub.](#)
- [5] [Miniconda — Conda documentation](#) (z tej strony można pobrać Minicondę)
- [6] [Managing environments — conda 4.9.2.post24+e37cf84a documentation](#) (Zarządzanie środowiskami w Minicondzie)
- [7] [Spyder :: Anaconda Cloud](#) (instalacja Spydera)
- [8] [Jupyterlab :: Anaconda Cloud](#) (instalacja Jupyter Lab)
- [9] [Zintegrowane środowisko programistyczne – Wikipedia, wolna encyklopedia](#)
- [10] [Getting Started With JupyterLab - DZone Big Data](#) (jeśli chcesz wiedzieć więcej o Jupyter Lab)
- [11] [Python Data Types \(programiz.com\)](#) (trochę o typach zmiennych)
- [12] [Python List Methods | Programiz](#) (metody list)
- [13] [Python String Methods | Programiz](#) (metody łańcuchów znaków)
- [14] [Python Dictionary \(With Examples\) \(programiz.com\)](#) (trochę o słownikach)
- [15] [Python Set Methods \(w3schools.com\)](#) (metody zbiorów)
- [16] [Python Set Methods | Programiz](#) (metody zbiorów)



SMARTFACTOR



+48 798 622 487



ul. Poselska 29
03-931 Warszawa



mail@smartfactor.pl



www.smartfactor.pl