

# Wizualizacja danych

Wykład 1

# Sprawy organizacyjne

# Sprawy organizacyjne

- Sylabus jest dostępny w systemie USOS.
- Regulamin zajęć dostępny jest na stronie prowadzącego zajęcia <http://wmii.uwm.edu.pl/~piojas/>.
- Forma zaliczenia: egzamin.
- Wykład - 15 godzin.
- <https://github.com/pjastr/WizualizacjaDanychStac2023>

# Wymagania wstępne

- Znajomość podstawowych konstrukcji programistycznych (ze wstępu do programowania).
- Matematyka z zakresu szkoły średniej/z przedmiotu repozytorium matematyki elementarnej.

Ewentualne braki należy opanować w samodzielny zakresie.

W razie problemów zapraszam na konsultacje.

# Wstęp do języka Python

# Język Python

- Poprawna wymowa: pajton.
- Język Python stworzył we wczesnych latach 90. Guido van Rossum – jako następcę języka ABC.
- Nazwa języka pochodzi od serialu komediowego emitowanego w latach siedemdziesiątych przez BBC – „Monty Python’s Flying Circus” (Latający cyrk Monty Pythona). Projektant, będąc fanem serialu i poszukując nazwy krótkiej, unikalnej i nieco tajemniczej, uznał tę za świetną.

# Przełomowy rok - 2008

- Utworzenie drugiej gałęzi rozwoju 3.x. Początkowe obie gałęzie były rozwijane niezależnie, lecz wsparcie Pythona 2.x zostało zakończone w roku 2020.
- Python 2.x cały czas jest wykorzystywany np. w ArcGis Desktop <https://support.esri.com/en/technical-article/000013224>

# Podstawowe różnice między 2.x a 3.x

- funkcja `print`

## Python 2:

```
1 print 'Hello, World!'
2 print('Hello, World!')
3 print "text", ; print 'print more text on the same line'
```

## Python 3

```
1 print('Hello, World!')
2 print("some text,", end='')
3 print(' print more text on the same line')
```



# Dzielenie zmiennych typu `int`

## Python 2:

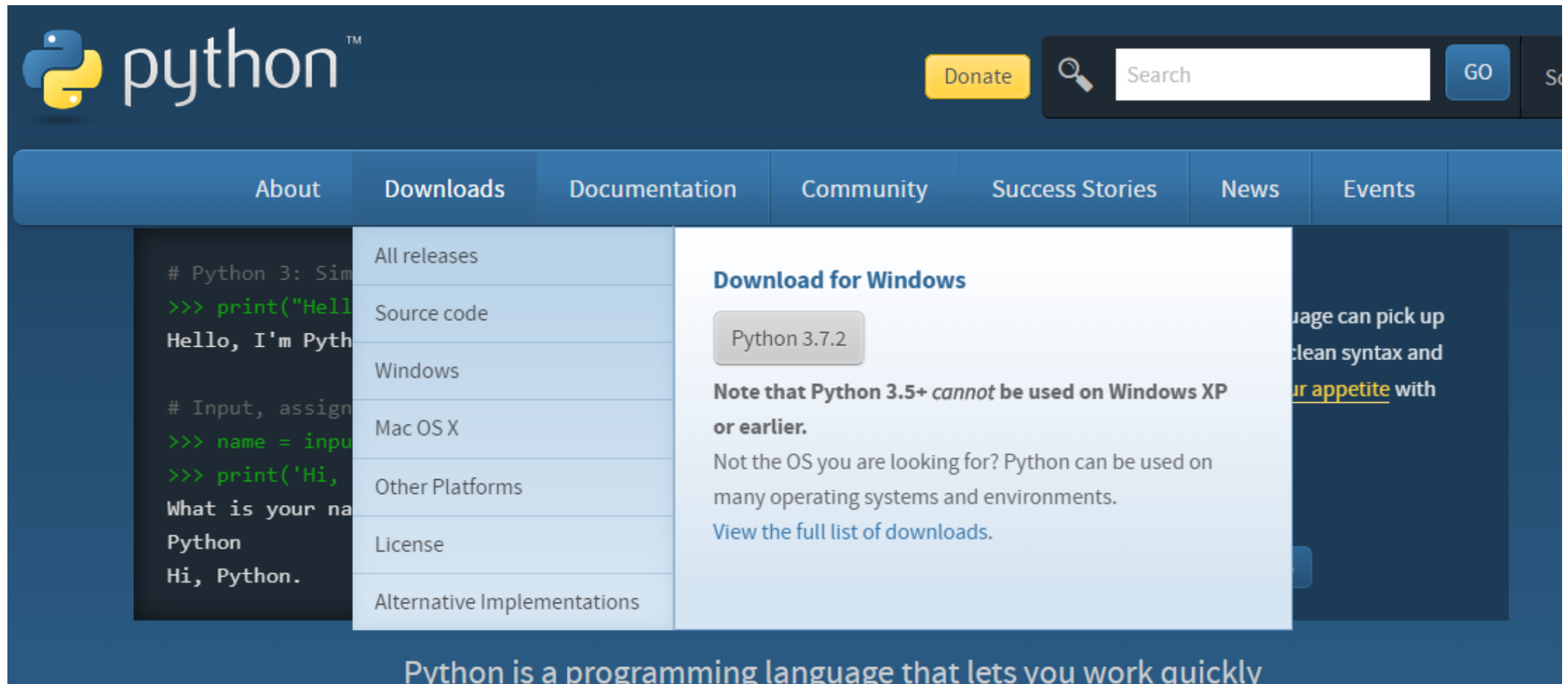
```
1 print '3 / 2 =', 3 / 2
2 print '3 // 2 =', 3 // 2
3 print '3 / 2.0 =', 3 / 2.0
4 print '3 // 2.0 =', 3 // 2.0
```

## Python 3

```
1 print('3 / 2 =', 3 / 2)
2 print('3 // 2 =', 3 // 2)
3 print('3 / 2.0 =', 3 / 2.0)
4 print('3 // 2.0 =', 3 // 2.0)
```

# Instalacja - Windows

- <https://python.org/>



The screenshot shows the Python.org website with the 'Downloads' menu open. The menu options are: All releases, Source code, Windows, Mac OS X, Other Platforms, License, and Alternative Implementations. The 'Download for Windows' section is highlighted, showing a button for 'Python 3.7.2'. A note states: 'Note that Python 3.5+ cannot be used on Windows XP or earlier.' Below this, it says: 'Not the OS you are looking for? Python can be used on many operating systems and environments. View the full list of downloads.'

python™

Donate

Search GO

About Downloads Documentation Community Success Stories News Events

```
# Python 3: Simple
>>> print("Hello, I'm Python")
Hello, I'm Python

# Input, assignment
>>> name = input("What is your name? ")
>>> print('Hi, Python')
What is your name? Python
Hi, Python.
```

**Download for Windows**

Python 3.7.2

**Note that Python 3.5+ cannot be used on Windows XP or earlier.**

Not the OS you are looking for? Python can be used on many operating systems and environments. [View the full list of downloads.](#)

Python is a programming language that lets you work quickly

# Linux

Sprawdzenie wersji na Ubuntu 18.04:

```
piotrekwd@piotrekwd-VirtualBox:~$ python3
Python 3.6.5 (default, Apr 1 2018, 05:46:30)
[GCC 7.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
```

Ubuntu

## Ręczna instalacja:

```
1 sudo apt install python3
```

# Wybór IDE do Pythona

- IDLE (domyślny)
- PyCharm <https://www.jetbrains.com/pycharm/> (na ćw. i wykład)
- Spyder IDE <https://www.spyder-ide.org/>
- Visual Studio <https://visualstudio.microsoft.com/pl/vs/features/python/>
- Visual Studio Code + odpowiednie rozszerzenia <https://code.visualstudio.com/>
- Atom + ide-python <https://atom.io/packages/ide-python>
- <https://colab.research.google.com/>
- <https://datalore.jetbrains.com/>
- i wiele innych...

# Styl PEP8

- wymowa: pi-i-pi-ejt
- standaryzacja kodu używana m.in. przy rozwijaniu nowych funkcjonalności
- używanie daje lepszą organizację i czytelność kod
- pełna wersja <https://www.python.org/dev/peps/pep-0008/>

## Znaki odstępu:

- we wcięciach stosujemy spacje (a nie tabulatory)
- każdy poziom wcięcia powinien składać się z 4 spacji
- wiersz powinien składać się z maksymalnie 79 znaków

## Puste linie:

- dwie linie między funkcjami najwyższego poziomu i między klasami.
- pojedyncza linia między funkcjami w klasie

## Kodowanie:

- dla Pythona 3 sugerowane i domyślne to UTF-8.

## Stringi:

- można używać pojedynczych apostrofów jak i podwójnych cudzysłówów
- ważne, aby stosować wybraną notację konsekwentnie
- jedyny wyjątek to gdy wewnątrz stringu chcemy użyć cudzysłów np.

```
1 print('Oglądam film "Player One"')
```



# Bibliografia

- <https://pl.wikipedia.org/wiki/Python>, dostęp online 12.02.2019.
- <https://bulldogjob.pl/news/264-java-php-ruby-jak-wlasciwie-wymawiac-nazwy-technologii>, dostęp online 12.02.2019.
- [https://sebastianraschka.com/Articles/2014\\_python\\_2\\\_3\\_key\\_diff.html](https://sebastianraschka.com/Articles/2014_python_2\_3_key_diff.html), dostęp online 14.02.2019.
- K. Ropiak, Wprowadzenie do języka Python, <http://wmii.uwm.edu.pl/~kropiak/wd/Wprowadzenie%20do%20j%C4%99zyka%20Python.p>, dostęp online 14.02.2019.
- B. Slatkin, Efektywny Python. 59 sposobów na lepszy kod, Helion 2015.
- <https://www.python.org/dev/peps/pep-0008/>, dostęp online 14.02.2019.

