

## Wizualizacja danych - wykład 2

## Cechy języka Python

- ▶ Python wspiera różne paradygmaty programowania: obiektowy, imperatywny oraz funkcyjny.
- ▶ Posiada w pełni dynamiczny system typów i automatyczne zarządzanie pamięcią (garbage collector).
- ▶ Często używany jako język skryptowy. Interpretery Pythona są dostępne na wiele systemów operacyjnych. Różne implementacje Pythona: CPython (język C), IronPython (platforma .NET), Jython (Java), PyPy (Python).
- ▶ Prosta i czytelna składnia ułatwiająca utrzymywanie, używanie i rozumienie kodu.

# Zen

```
import this
```

The Zen of Python, by Tim Peters

Beautiful is better than ugly.

Explicit is better than implicit.

Simple is better than complex.

Complex is better than complicated.

Flat is better than nested.

...

PL - [https:](https://pl.python.org/)

[//pl.python.org/forum/index.php?topic=392.msg1844#msg1844](https://pl.python.org/forum/index.php?topic=392.msg1844#msg1844)

# Zmienna

- ▶ najprościej: przechowuje pewną wartość:

```
x = 5
```

```
y = "John"
```

# Wbudowane typy danych

- ▶ numeryczne (liczbowe): int, float, complex
- ▶ tekstowe: str
- ▶ sekwencji: list, tuple
- ▶ odwzorowania (mapping type): dict
- ▶ zestawów (set types): set, frozenset
- ▶ logiczne: bool
- ▶ binarne: bytes, bytearray

## Int - typ całkowity

- ▶ bez kropki dziesiętnej
- ▶ może być dowolnie długi (ograniczenie ilość pamięci)

```
print(123123123123123123123123123123123123123123 + 1)
```

## Jaki system liczbowy?

Domyślnie dziesiętny. Więcej za kilka zajęć przy algorytmach liczbowych.

```
print(101)
print(101)
print(0x101) # zero-x
print(0o101) # zero-litera o
print(0b101) # zero-b
print(OX101) # zero-x
print(OO101) # zero-litera o
print(OB101) # zero-b
```

## Sprawdzenie typu

```
print(type(234))
```

<class 'int'> - wszystko jest obiektem



## Float - typ zmiennoprzecinkowy

```
print(4.2)
print(4.)
print(.5)
print(.3e4)
print(3e4)
print(2e-2)
print(1.79e308)
print(1.8e308)
print(5e-324)
print(1e-325)
```

## Complex - typ zespolony (dot. liczb zespolonych)

```
print(2+3j)  
print(2+5*1j)
```

Uwaga: wyrażenie

```
print(4+5*j)
```

skutkuje błędem w większości wypadków.

str - string, napisy, łańcuchy znaków

Obecnie odchodzi się od określenia “tablica znaków”

```
print("ABC")  
print('abc')
```

## bool - typ logiczny

```
print(True)  
print(False)
```

## Operator przypisania

= przypisuje prawą stronę do lewej (!), często mylony z operatorem logicznym równa się ==

```
x = 5
```

```
x = "Piotr"
```

```
a = 4.5
```

```
A = 56
```

```
x, y, z = "Orange", "Banana", "Cherry"
```

```
x = y = z = "Orange"
```

## input - operacja wejścia

```
num = input ("Wprowadź liczbę :")  
print(num)  
name1 = input("Wprowadź imię : ")  
print(name1)
```

- ▶ zawsze przyjmuje napis, w razie potrzeby trzeba zrzutować

```
x = str(num)  
y = int(num)  
z = float(num)
```

## print - instrukcja wyjścia

```
print(4.2)
# print('Mój wiek to' + 36)
print('Mój wiek to', 36)
a = 36
print('Mój wiek to', a)
print('hello', 'world')
print('hello', 'world', sep='')
print('hello', 'world', sep='\n')
```

```
print(*objects, sep=' ', end='\n', file=sys.stdout,  
      flush=False)
```

- ▶ `objects` - to co ma być wyświetlone
- ▶ `sep` - separator, domyślnie znak spacji
- ▶ `end` - co co ma być wyświetlone na końcu, domyślnie znak końca linii
- ▶ `file` - określa gdzie mają być `objects` wyświetlone, domyślnie `sys.stdout` (domyślny ekran)
- ▶ `flush` - określa czy “wyjście” ma być buforowane przed przekazaniem do `file`, domyślne `False`



```
print(1, 2, 3, 4)
```

```
## 1 2 3 4
```

```
print(1, 2, 3, 4, sep='*')
```

```
## 1*2*3*4
```

```
print(1, 2, 3, 4 ,sep='#', end='&')
```

```
## 1#2#3#4&
```

```
print('x', 'y', 'z', sep='', end='')  
print('a', 'b', 'c', sep='', end='')
```

```
## xyzabc
```

```
print('a', 'b', '\n', 'c')
```

```
## a b
```

```
## c
```

\t - przesunięcie do następnego "tab"=8 spacji

```
print('sdf', 3456, -2, sep='\t')
```

```
## sdf    3456    -2
```

Formatowanie napisów będzie później.

## Operacje arytmetyczne

Operator	Opis	Składnia
+	Dodawanie	$x + y$
-	Odejmowanie	$x - y$
*	Mnożenie	$x * y$
/	Dzielenie	$x / y$
//	Dzielenie całkowite	$x // y$
%	Dzielenie modulo	$x \% y$
**	Potęgowanie	$x ** y$

```
print(5+3)
```

```
## 8
```

```
print(4*5.2)
```

```
## 20.8
```

```
print(9-7)
```

```
print(4/5)
```

```
## 0.8
```

```
print(4//5)
```

```
## 0
```

```
print(4/5.0)
```

```
## 0.8
```

```
print(4//5.0)
```

```
## 0.0
```

```
print(3**0)
```

```
## 1
```

```
print(0**0)
```

```
## 1
```

```
print(4/0)
```

ZeroDivisionError: division by zero

## Przypisanie z operacją arytmetyczną

Lista zawiera wybrane operacje.

Inna nazwa to złożone operatory przypisania.

Operator	Zapis	Dłuższa wersja
<code>+=</code>	<code>x += 5</code>	<code>x = x + 5</code>
<code>-=</code>	<code>x -= 5</code>	<code>x = x - 5</code>
<code>*=</code>	<code>x *= 5</code>	<code>x = x * 5</code>
<code>/=</code>	<code>x /= 5</code>	<code>x = x / 5</code>
<code>%=</code>	<code>x %= 5</code>	<code>x = x % 5</code>
<code>//=</code>	<code>x //= 5</code>	<code>x = x // 5</code>
<code>**=</code>	<code>x **= 5</code>	<code>x = x ** 5</code>

```
a = 5
a += 1
print(a)
```

## 6

```
a **= 2
print(a)
```

## 36



## Bibliografia

- ▶ <https://pl.wikipedia.org/wiki/Python>, dostęp online 12.02.2019.
- ▶ <https://bulldogjob.pl/news/264-java-php-ruby-jak-wlasciwie-wymawiac-nazwy-technologiei>. 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.pdf>, 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.

## Bibliografia - cd2

- ▶ <https://www.flynerd.pl/2017/05/python-4-typy-i-zmienne.html>, dostęp online 14.02.2019.
- ▶ <http://pytolearn.csd.auth.gr/p0-py/01/print.html>, dostęp online 15.02.2019.
- ▶ [https://www.tutorialspoint.com/python3/python\\_lists.htm](https://www.tutorialspoint.com/python3/python_lists.htm), dostęp online 17.02.2019.

## Bibliografia - cd3

- ▶ <https://realpython.com/python-data-types/>, dostęp online 5.01.2022
- ▶ [https://www.w3schools.com/python/python\\_variables.asp](https://www.w3schools.com/python/python_variables.asp), dostęp online 5.01.2022
- ▶ [https://www.w3schools.com/python/python\\_variables\\_multiple.asp](https://www.w3schools.com/python/python_variables_multiple.asp), dostęp online 5.01.2022
- ▶ <https://realpython.com/python-print/>, dostęp online 5.01.2022
- ▶ <https://www.programiz.com/python-programming/operators>, dostęp online 5.01.2022
- ▶ <https://realpython.com/python-conditional-statements/>, dostęp online 5.01.2022
- ▶ <https://realpython.com/python-for-loop/>, dostęp online 5.01.2022
- ▶ <https://realpython.com/python-while-loop/>, dostęp online 5.01.2022