

Podstawy Python 1

Wojciech Barczyński

Spis treści

Spis treści	2
1. Pierwszy program	3
2. Typy Danych	3
3. Pierwsza funkcja	7

1. Pierwszy program

Nasz pierwszy program, naturalnie Hello World.

1. Przygotujemy nasz projekt:

```
$ mkdir -p GIT_USERNAME/nauka_pythona  
$ cd GIT_USERNAME/nauka_pythona
```

2. Utwórz plik main.py :

```
print("Hello World!")
```

Uruchom twój pierwszy program:

```
python main.py
```

2. Typy Danych

1. Wprowadzenie do zmiennych 1

```
imie = 'Wiktoria'  
print("Hello World!" + imie)
```

2. Wprowadzenie do zmiennych 2

```
imie = 'Wiktoria'  
ilosc_ksiazek = 10  
srednia = 4.5  
  
print("Hello World!" + imie)  
print("Ma " + str(ilosc_ksiazek))  
print("Srednia: " + str(srednia))
```

Zmodyfikuj powyższy program, aby był o czymś innym, np.: samochodach, samolotach.

Nazewnictwo:

- camelCase
- snake_case

3. Kalkulator, środowisko python jest doskonałe do obliczeń:

```
$ python
>>> 1 + 3
# uwaga!:
>>> 10 / 3
>>> 10.0 / 3
>>> float(10) / 3
# ze zlotówek na Euro
>>> 4000 / 4.3
# 30%
>>> (4000.0 * 33) / 100
# zmienne też działają
>>> cena_euro = (4000.0 ) / 4.3
>>> ( cena_euro * 33 ) / 1000
>>> cena_euro
```

Zadanie domowe: przez najbliższy tydzień wykorzystaj pythona do szybkich obliczeń.

4. Krótko o IPython:

```
$ ipython
$ print('jak w python ale wspiera komendy bash')
$ ! ls /home/tester
$ moj_home = ! ls /home/tester
$ moj_home
```

Wskazówka: Warto poświęcić czas, aby poznać bliżej IPython.

5. Funkcje string:

```
marka = 'Pegout'
ilosc_drzwi = 5
pojemność = 1.3

marka_up = marka.upper()
```

```
print("Samochod " + marka + " ma " + str(ilosc_drzwi) + " dzwi")
print(marka_up)
print("Pojemnosc po zmianach: " + str(pojemnosc * 2))
```

Korzystając z dokumentacji, znajdź funkcję, która zmieni wszystkie litery w nazwie samochodu na małe litery.

Wykorzystaj dodatkowo dowolną funkcję, którą znajdziesz w dokumentacji.

6. Czytelniejszy kod z *string.format* :

```
imie = 'Ala'
zwierze = 'kot'
print("{0} ma {1}a".format(imie, zwierze))
```

7. Instrukcje Warunkowe.

Jak sprawdzić, czy string zaczyna się od *Pe*.

8. Listy 1:

```
samochody = ['syrena', 'poloneza']

print(samochody[0])
print(samochody[1])

for s in samochody:
    print(s)

for idx in range( len(samochody) ) :
    print("idx: " + str(idx) + " : " + samochody[idx])
```

W osobnym pliku napisz program o czymś innym, np., zwierzętach, samolotach, itp.

Co się stanie, gdy dodasz następującą linie do swojego programu:

```
print(samochody[3])
```

9. Listy i string:

```
print(",".join(samochod))
arr = "a,b,c,d,e".split(',')
```

10. Kasowanie i zmiany elementów w liście:

```
del samochod[0]
samochod[1] = 'kia'
```

11. Listy 2:

```
samochody = ['syrena', 'poloneza']
ilosc = [3, 5]

for idx in range( len(samochody) ) :
    print("idx: " + str(idx) + ": " + samochody[idx])
    print(samochody[idx] + " ma ilosc drzwi " + str(ilosc[idx]))
```

W osobnym pliku napisz program o czymś innym, np., zwierzętach, samolotach, itp.

12. Dodatkowe – koszyk z regułami na obniżki. Zamodeluj koszyk jako dwie listy – jedna z produktami, a druga z cenami.

- R1: Więcej niż 3 rzeczy 5 % zniżki
- R2: Wartość koszyka 500 PLN 10% zniżki

Dodatkowe 1: Jeśli użyliśmy obniżki według R1 to nie używamy reguły według R2.

Dodatkowe 2: Jeśli R2 to nie R1.

Dodatkowe 3: R3 – Najtańszy produkt jest gratis.

Dodatkowe 4: R4 – co trzeci produkt gratis

13. Dictionary 1: możemy reprezentować w sposób nieustrukturyzowany obiekty.

```
samolot = {'name': 'boeing',
            'przebieg': 10000,
            'type': 'pasazerski'}

print(samolot['name'])
print(samolot['type'])

# co wypiszemy na ekran
print(samolot['nieznany_klucz'])
# a co, kiedy?
print(samolot.get('nieznany_klucz'))

# in python3 samolot.items()
for key, value in samolot.items():
    print("{0}:{1}".format(key, value))

#
for key in samolot:
    print("{0}:{1}".format(key, samolot[key]))
```

Dodatkowe 1: Zamodeluj nasz koszyk z pomocą słownika, zsumuj jego wartość. Każda pozycja koszyka ma *nazwę* i *wartość*.

Dodatkowe 2: Jeśli jest ser i mleko w koszyku, cena -10%.

Wariacja: obliczanie liczby punktów w grze albo obrażeń.

14. Dictionary 2: lista z identyfikatorami.

```
produkty = {'S1222': 'sukienka trojkat',
            'P1222': 'spodnie krata',
            'X212': 'konsola do gier'}

igla = 'X2X'

if igla in produkty:
    print("Znalazlem {0}".format(igla))
else:
    print("Brak w magazynie {0}".format(igla))
```

3. Pierwsza funkcja

Napiszmy naszą własną funkcję do wypisywania typów danych dictionary.

```
def print_dict(d):
    for key in samolot:
        print("{0}:{1}".format(key, d[key]))

if __name__ == "__main__":
    samolot = {'name': 'boeing',
              'przebieg': 10000,
              'type': 'pasazerski'}

    print_dict(samolot)
    print_dict(samolot)
```