

## Typy danych

- string – łańcuchy znaków, zapisujemy pomiędzy znakami “ lub "" (muszą być zapisane w jednym wierszu) albo potrójnymi apostrofami (‘’) czy potrójnymi cudzysłowami (’’’) - mogą być pisane w kilku liniach.

np. a = 'Ala ma kota'

```
a = ""Ala
ma
dużego
kota""
```

```
a = 'Ala\nma\ndużego\nkota'
```

- integer - całkowite
- float - rzeczywiste (zmiennoprzecinkowe, separatorem dziesiętnym jest .)
- complex – zespolone (liczby postaci a+bj)

Komentarze piszemy po znaku #, jeżeli chcemy automatycznie za komentować kilka linii istniejącego kodu stosujemy kombinacje klawiszy ctrl+/ (tej samej kombinacji klawiszy stosujemy jak chcemy usunąć komentarz z kilku linii).

## Zmienne

### Deklaracja

nazwa\_zmiennej = wartość

### Usuwanie

del a #usuwa zmienna

### Drukowanie

```
print(nazwa_zmiennej) #drukuje zmienną
print(id(nazwa_zmiennej)) #drukuje adres zmiennej
```

### Deklaracja wielokrotna

zm1, zm2, zm3, ..., zmn = wart1, wart2, wart3, ..., wartn

## Zasady tworzenia zmiennych

Możemy używać „podkreślenia” czyli „\_” ale nie wolno używać „minusa” czyli „-”.

Nazwa nie może zaczynać się od cyfry. Cyfry mogą się pojawić w dalszej części nazwy

Dla lepszej czytelności używamy małych liter po znaku „\_”

## Przykłady tworzenia zmiennych.

```
a = '123456' #to jest łańcuch
b = '$zmienna' #to też jest łańcuch
print(a+b)
c,d = 2, 3.14 #wielokrotna deklaracja zmiennych
wynik = c + d
print(wynik)
e = 3 + 2j
print(e)
```

## Działania arytmetyczne

### Przykłady działań arytmetycznych

```
a = 8
b = 4
c = 3
dzielenie = a / b
print(dzielenie)
dzielenie = a / c
print(dzielenie)
dodawanie = a + b
print(dodawanie)
dzielenie_calkowite = a // c
print(dzielenie_calkowite)
reszta = a % c
print(reszta)
potega = b ** c
print(potega)
potega = pow(4, 3)
print(potega)
```

## Operatory przyrostkowe

```
a = 8
#zamiast pisać a=a+2 można zapisać
a += 2
print(a)
a -= 2
print(a)
a *= 2
print(a)
a **= 2
print(a)
a /= 2
print(a)
a //= 5
print(a)
```

## Formatowanie łańcuchów podczas wyświetlania

#Drukujemy liczby

```
print('a=%(zm)d' % {'zm':12})
```

```
a = 5
```

```
b = 3
```

```
z = a - b
```

```
print('Wynik działania %(z1)d-%(z2)d=%(z3)d' % {'z1':a, 'z2':b, 'z3':z})
```

zm, z1, z2, z3 to są nazwy zmiennych, które będą formatowane i pod które można podstawiać odpowiednie liczby

Inny sposób

```
print('a={0:d}'.format(12))
```

```
a = 5
```

```
b = 3
```

```
z = a - b
```

```
print('Wynik działania {0:d} - {1:d} = {2:d}'.format(a, b, z))
```