

Coś tam o obiektach



Obiekt charakteryzuje się:



- Dane / stan
 - Zmienne / pola przechowywane w obiekcie
- Zachowanie
 - Metody / czynności, które obiekt potrafi wykonać
 - Co prawda w klasie jest jedna definicja (każdy obiekt wykonuje ten sam kod), ale każdy odwołuje się do własnych zmiennych



Klasa jest wzorcem dla obiektów



- Programista tworzy klasę
 - To jest po prostu ten kod
- Obiekty powstają w czasie działania programu
- Zazwyczaj w programie istnieje wiele obiektów



Zły obrazek



Tak **NIE JEST** w Pythonie!

a:

Konto

- nr: 1
- saldo: 1000
- wł.: Ala

b:

Konto

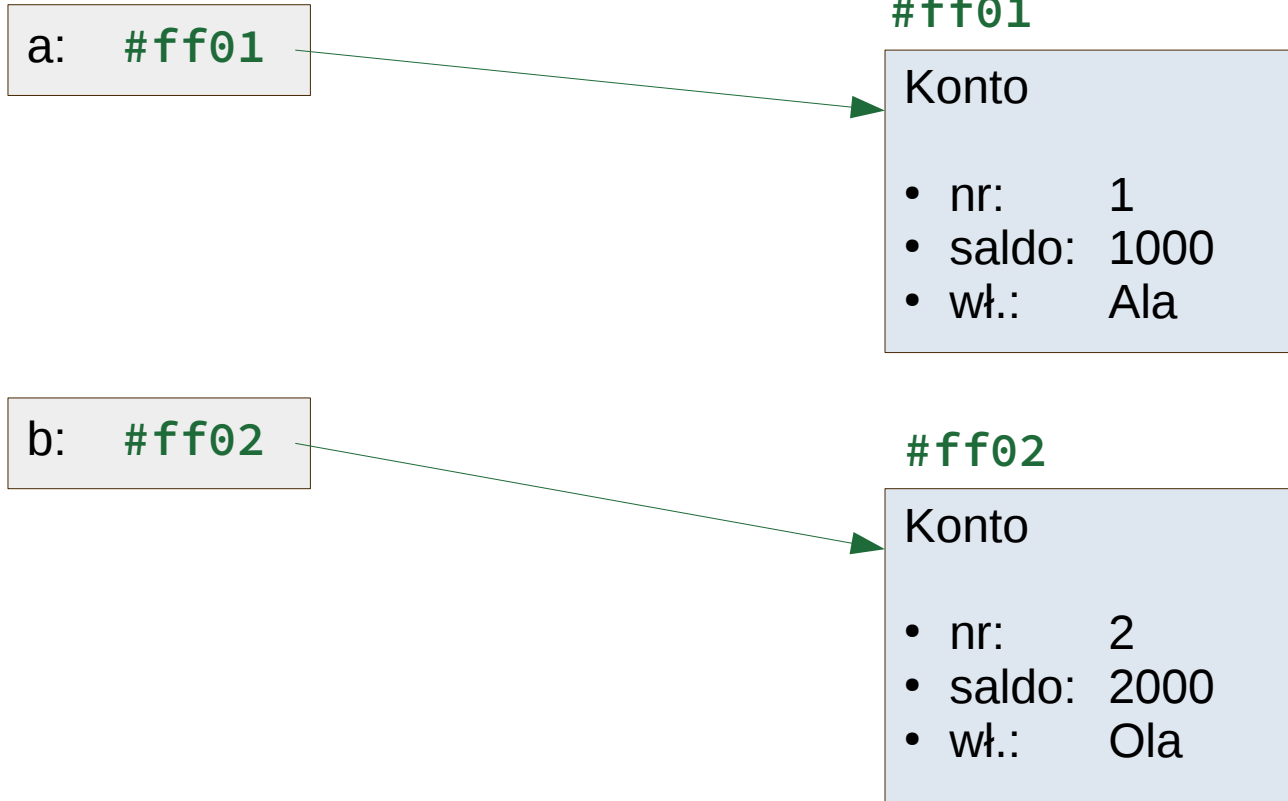
- nr: 2
- saldo: 2000
- wł.: Ola

Referencje 1 Krok 1



zmienne

obiekty

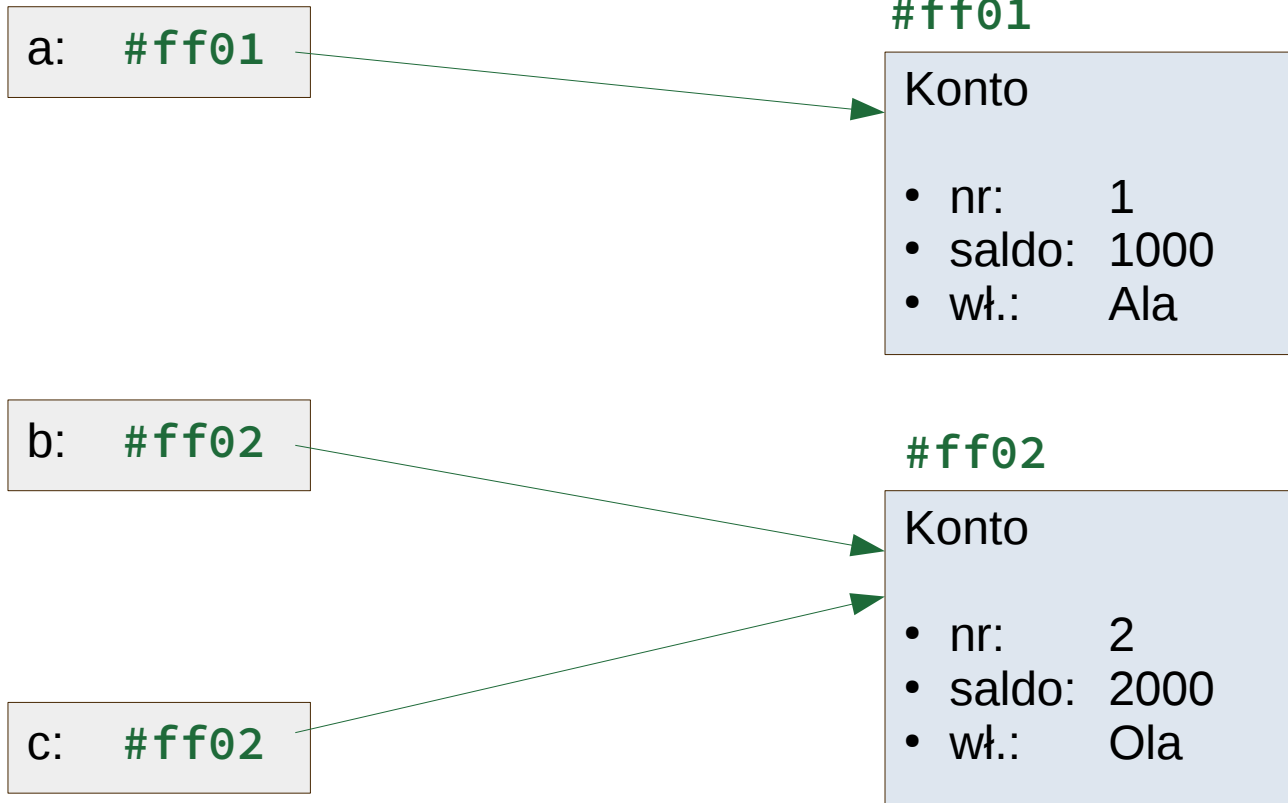


Referencje 1 Krok 2



zmienne

obiekty



Referencje 1 Krok 3



zmienne

obiekty

a: #ff01

#ff01

Konto

- nr: 1
- saldo: 1000
- wł.: Ala

operacje:
b.wplata(40)
c.wplata(8)

b: #ff02

#ff02

Konto

- nr: 2
- saldo: 2048
- wł.: Ola

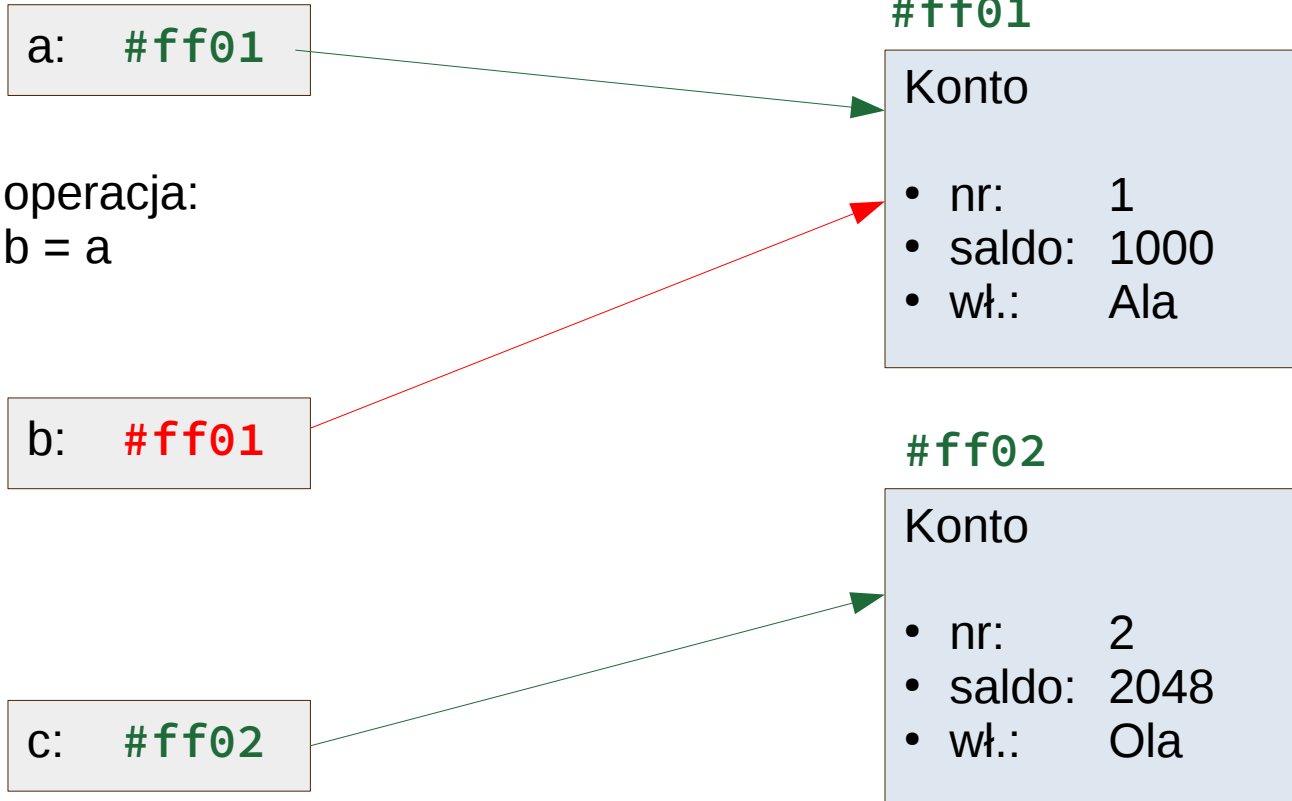
c: #ff02

Referencje 1 Krok 4



zmienne

obiekty

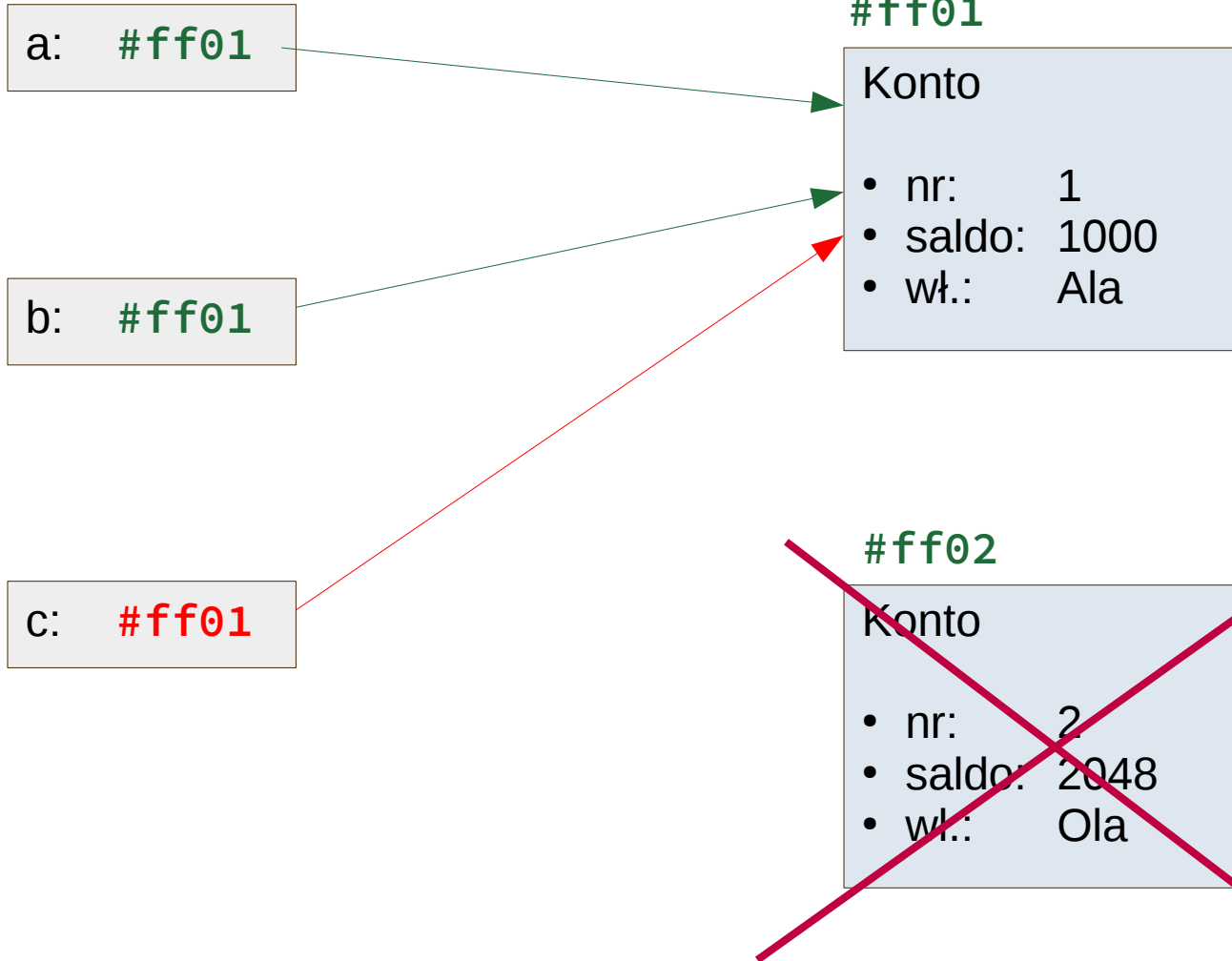


Referencje 1 Krok 5



zmienne

obiekty



garbage collector
może usunąć
nieużywany obiekt,
ale zwykle robi dopiero,
gdy zaczyna brakować pamięci

Referencje 1 Krok 6



zmienne

a: **NULL**

b: **#ff01**

c: **#ff01**

obiekty

#ff01

Konto

- nr: 1
- saldo: 1000
- wł.: Ala

#ff02

Konto

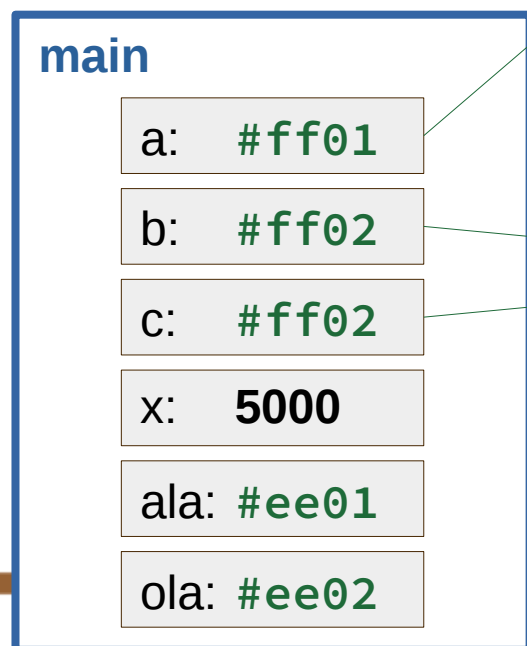
- nr: 2
- saldo: 2048
- wł.: Ola

garbage collector
może usunąć
nieużywany obiekt,
ale zwykle robi dopiero,
gdy zaczyna brakować pamięci

stos / stack

Referencje 2 Krok 1

sterta / heap



#ff01

Konto

- nr: 1
- saldo: 1000
- wł.: Ala

#ff02

Konto

- nr: 2
- saldo: 2000
- wł.: Ola

#ee01

Osoba

- imie: Ala
- nazw Kow

#ee02

Osoba

- imie: Ola
- nazw Mal

stos / stack

Referencje 2 Krok 2

sterta / heap



funkcja

a: #ff01

b: #ff02

c: #ff02

x: 5000

#ff01

Konto

- nr: 1
- saldo: 1000
- wł.: Ala

#ee01

Osoba

- imie: Ala
- nazw: Kow

main

a: #ff01

b: #ff02

c: #ff02

x: 5000

ala: #ee01

ola: #ee02

#ff02

Konto

- nr: 2
- saldo: 2000
- wł.: Ola

#ee02

Osoba

- imie: Ola
- nazw: Mal

stos / stack

Referencje 2 Krok 3

sterta / heap



funkcja

a: #ff01

b: #ff02

c: #ff02

x: 5055

#ff01

Konto

- nr: 1
- saldo: 1000
- wł.: Ala

#ee01

Osoba

- imie: Ala
- nazw Kow

main

a: #ff01

b: #ff02

c: #ff02

x: 5000

ala: #ee01

ola: #ee02

#ff02

Konto

- nr: 2
- saldo: 2000
- wł.: Ola

#ee02

Osoba

- imie: Ola
- nazw Mal

stos / stack

Referencje 2 Krok 4

sterta / heap



funkcja

a: #ff01

b: #ff02

c: #ff02

x: 5055

#ff01

Konto

- nr: 1
- saldo: 1000
- wł.: Ala

#ee01

Osoba

- imie: Ala
- nazw: Kow

main

a: #ff01

b: #ff02

c: #ff02

x: 5000

ala: #ee01

ola: #ee02

#ff02

Konto

- nr: 2
- saldo: 2044
- wł.: Ola

#ee02

Osoba

- imie: Ola
- nazw: Mal

stos / stack

Referencje 2 Krok 5

sterta / heap



#ff03

Konto

- nr: 1
- saldo: 1000
- wł.: Ala

#ff01

Konto

- nr: 1
- saldo: 1000
- wł.: Ala

#ee01

Osoba

- imie: Ala
- nazw Kow

#ff02

Konto

- nr: 2
- saldo: 2044
- wł.: Ola

#ee02

Osoba

- imie: Ola
- nazw Mal

funkcja

a: #ff03

b: #ff02

c: #ff02

x: 5055

main

a: #ff01

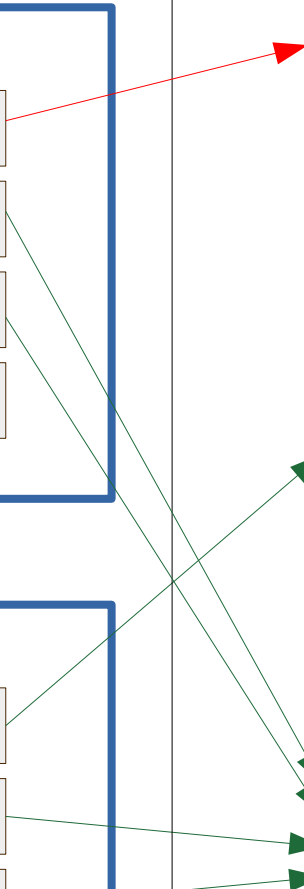
b: #ff02

c: #ff02

x: 5000

ala: #ee01

ola: #ee02



stos / stack

Referencje 2 Krok 6

sterta / heap



#ff03

Konto

- nr: 1
- saldo: 1024
- wł.: Ala

#ff01

Konto

- nr: 1
- saldo: 1000
- wł.: Ala

#ee01

Osoba

- imie: Ala
- nazw: Kow

#ff02

Konto

- nr: 2
- saldo: 2044
- wł.:

#ee02

Osoba

- imie: Ola
- nazw: Mal

funkcja

a: #ff03

b: #ff02

c: #ff02

x: 5055

main

a: #ff01

b: #ff02

c: #ff02

x: 5000

ala: #ee01

ola: #ee02

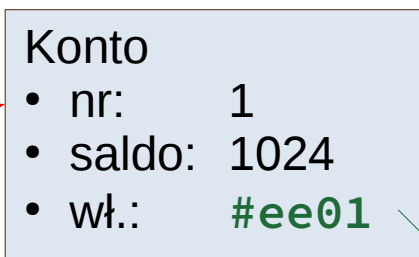
stos / stack

Referencje 2 Krok 7

sterta / heap



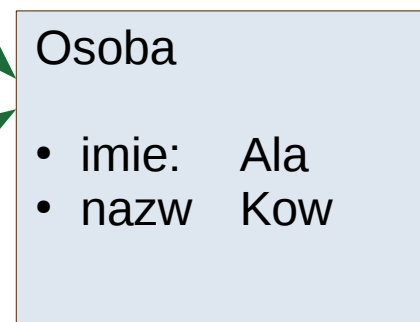
#ff03



#ff01



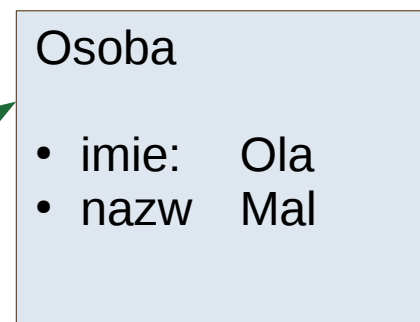
#ee01



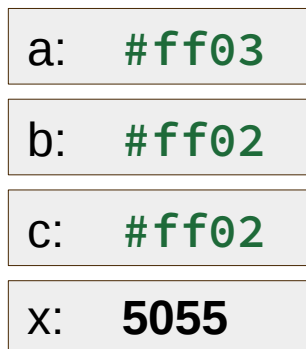
#ff02



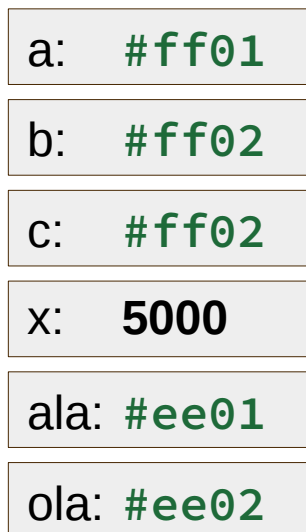
#ee02



funkcja



main



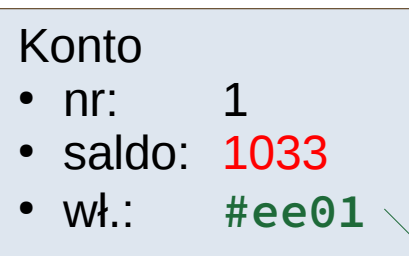
stos / stack

Referencje 2 Krok 8

sterta / heap



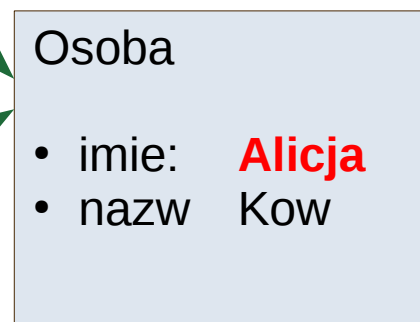
#ff03



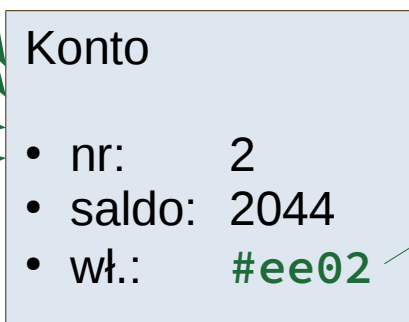
#ff01



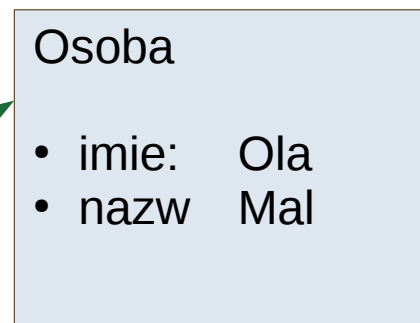
#ee01



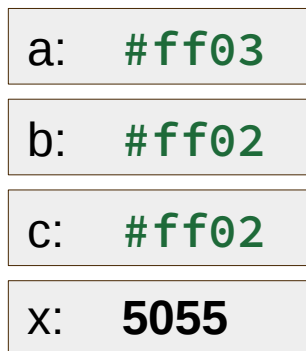
#ff02



#ee02



test



main

