

Tema 16

KLASE

Fabrika

OBJEKTI

Fiat 500, Fiat 500L



Nevidljivi objekat klase
self

Metoda za stvaranje objekata
init()

Delovi klase
Atributi i metode

-> Napraviti glavnu klasu Osoba

-> Dodati metodu pozdrav()

-> Dodati jos atributa i jos metoda za dohvatanje i setovanje tih atributa

PRIMER 1: PRAVIMO NOVI TIP PODATKA

```
class Osoba:
    def __init__(self, ime, godine):
        self.ime = ime
        self.godine = godine

    def pozdrav(self):
        print("Cao! Ja sam {} i imam {} godina!".format(self.ime, self.godine))

p1 = Osoba("Hana", 12)

print(p1.ime)
print(p1.godine)

p1.pozdrav()

del p1.godine

del p1

print(p1.ime)
```



ZADATAK 1

UCENICI U ALGORITMICI

Napraviti mini bazu učenika
Algoritmike, (max 5 učenika):
-> napraviti klasu Ucenik,
-> napraviti da se čuvaju u listi
-> napraviti da se ispisuje srednja
visina
-> napraviti da se ispisuje najveća
posećenost kursa.

```
ucenici = []

class Ucenik:
    def __init__(self, ime, razred, kurs, visina):
        self.ime = ime
        self.razred = razred
        self.kurs = kurs
        self.visina = visina

    def pozdrav(self):
        print("Ja sam {}, idem u {} razred i na {}
programiranje!".format(self.ime, self.razred, self.kurs))

    def mojaVisina(self):
        return int(self.visina)

    def mojKurs(self):
        return self.kurs

def unesi_ucenike():
    i = 0
    while i < 5:
        ucenik = input("Unesi ime:razred:kurs:visina za ucenika: ")
        (ime, razred, kurs, visina) = ucenik.split(':')
        u = Ucenik(ime, razred, kurs, visina)
        ucenici.append(u)
        i+=1

def ispisi_ucenike():
    for element in ucenici:
        element.pozdrav()

def srednjaVisina():
    suma = 0
```

```
for ucenik in ucenici:
    suma += ucenik.mojVisina()
print("Srednja visina {:.2f}".format(suma/len(ucenici)))

def najviseNaKursu():
    mapa = {}
    # put elements into map
    for element in ucenici:
        if element.mojKurs() not in mapa.keys():
            mapa[element.mojKurs()] = 1
        else:
            mapa[element.mojKurs()] += 1
    # find out highest value of key values
    max_ucenika = max(mapa.values())
    k = ""
    for kurs, broj in mapa.items():
        if broj == max_ucenika:
            k = kurs
            break
    # return that as result
    return max_ucenika, k

def main():
    unesi_ucenike()
    ispisi_ucenike()
    srednjaVisina()
    print(najviseNaKursu())

main()
```

PRIMER 3: KLASA ZIVOTINJA

```
class Zivotinja:
    broj_zivotinja = 0
    def __init__(self, ime):
        self.ime = ime
        Zivotinja.broj_zivotinja += 1

    def oglasi_se(self):
        print("Masem repom!")

    def pozdrav(self):
        print("Ja sam", self.ime)

class Slon(Zivotinja):
    def masemSurlom(self):
        print("fijuuu fijuuuu")

    def oglasi_se(self):
        Zivotinja.oglas_i_se(self)
        print("uuuuuuu")

class Majmun(Zivotinja):
    def igramSe(self):
        print(self.ime, "jede bananu.")
        for i in range(5):
            print("Skok", i)

    def oglasi_se(self):
        print("Mmmmmmm, mmmmmmm")

class Koza(Zivotinja):
    def oglasi_se(self):
        Zivotinja.oglas_i_se(self)
        print("meeee, me, meeeee")

    def skakucem(self):
        print("hop hop hop")

class Pingvin(Zivotinja):
    def oglasi_se(self):
        print("Sljap, sljap")

    def plivam(self):
        print("brck, brck, brck")

class Sova(Zivotinja):
    def oglasi_se(self):
        print("Hu hu huuuuu")

    def ukupim_se(self):
        print("ssss, ssss, ssss")
```

```
def main():
    print("Predstavljamo vam zivotinje u zoo vrtu: ")
    zivotinje = [Sova("Mile"), Pingvin("Pigo"), Koza("Rajka"), Majmun("Micko"),
Slon("Ciro") ]

    print("Ukupno ih ima:", Zivotinja.broj_zivotinja)

    print("Ovo je njihov pozdrav:")
    for element in zivotinje:
        element.pozdrav()
        element.oglasise()

main()
```