

Python - zadaci za vežbu iz OOP

A decorative graphic consisting of several horizontal lines of varying lengths and colors (teal and white) extending from the right side of the title area across the top of the slide.

Zadatak 1:

- Napisati klasu Automobil koja kreira podatke o automobilu. Dovoljan broj informacija o automobilu su naziv proizvođača, modela i maksimalna brzina kojom se automobil kreće. Klasa treba sadržati funkcije koje će upisati i odštampati podatke.

Zadatak 1 rešenje:

```
>>> class Automobil:
    # promenljive
    proizvodjac = "Nije unet proizvodjac"
    model = "Nije unet model"
    maxBrzina = 0

    # metode
    def nazivProizvodjaca(self, x):
        self.proizvodjac = x

    def postaviMaksimalnuBrzinu(self, x):
        self.maxBrzina = x
```

Zadatak 1 rešenje:

```
def postaviMaksimalnuBrzinu(self,x):  
    self.maxBrzina = x  
  
def postaviModel(self,x):  
    self.model = x  
  
def podaci(self):  
    print("Auto je proizveden u kompaniji", self.proizvodjac, ",")  
    print("model je", self.model, ",")  
    print("a maksimalna brzina je", self.maxBrzina, "km/h.")
```

```
>>> auto = Automobil()  
>>> auto.nazivProizvodjaca('zastava')  
>>> auto.postaviModel('yugo 45')  
>>> auto.postaviMaksimalnuBrzinu('180')  
>>> auto.podaci()  
Auto je proizveden u kompaniji zastava ,  
model je yugo 45 ,  
a maksimalna brzina je 180 km/h.
```

Zadatak 2:

- Napraviti klasu `OtvoriRacun` koja ima metode za uplatu, isplatu novca i proveru stanja računa korisnika.

Zadatak 2 rešenje:

```
>>> class OtvoriRacun:
    def __init__(self):
        self.stanje = 0

    def isplata(self, kolicina):
        if (self.stanje - kolicina < 0):

            print('Nemate dovoljno novca na racunu')
            return;

        else:
            self.stanje = self.stanje - kolicina

    def uplata(self, kolicina):
        self.stanje = self.stanje + kolicina

    def stanjeRacuna(self):
        print ((str)(self.stanje))
```

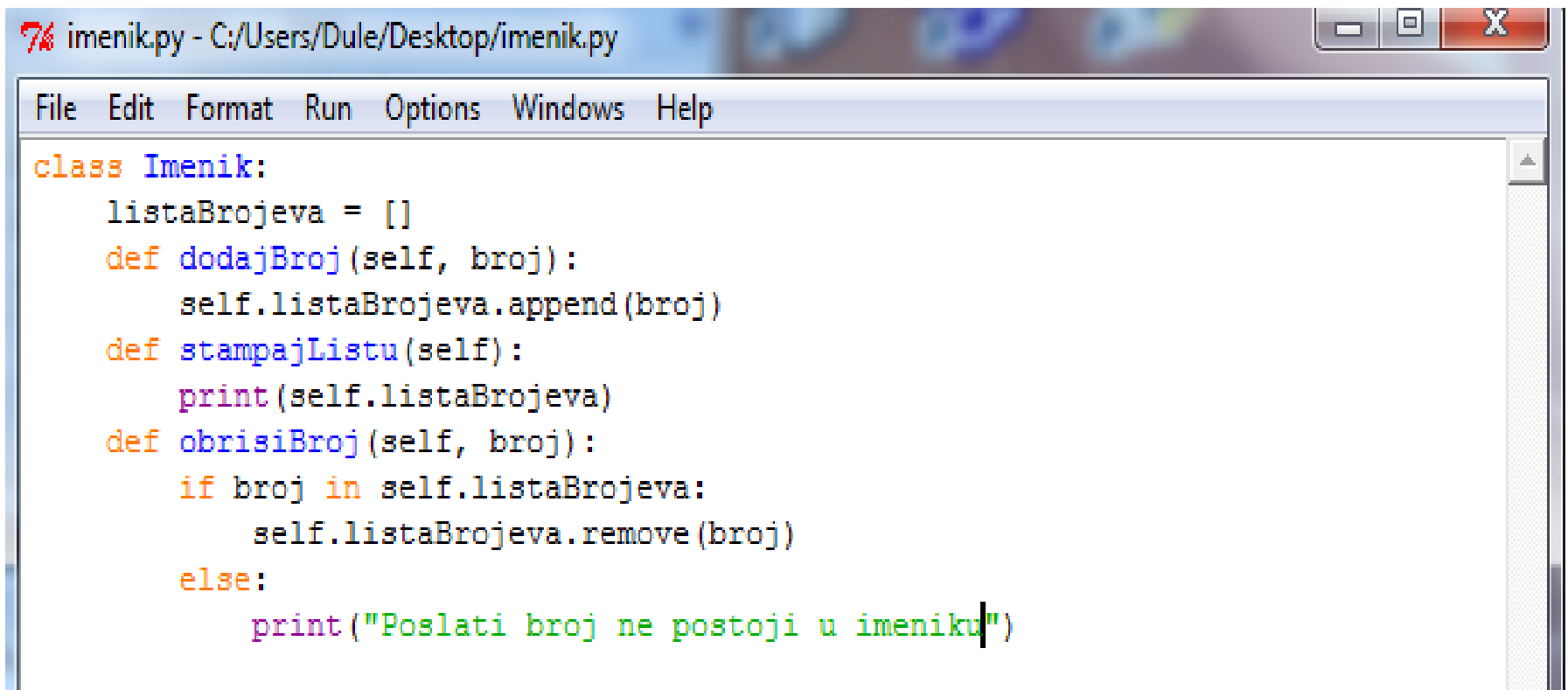
Zadatak 2 rešenje:

```
>>> marko = OtvoriRacun()
>>> marko.stanjeRacuna()
0
>>> marko.uplata(150)
>>> marko.stanjeRacuna()
150
.
>>> marko.isplata(50)
>>> marko.stanjeRacuna()
100
>>> marko.isplata(500)
Nemate dovoljno novca na racunu
```

Zadatak 3:

- Napisati program koji nam omogućava vođenje imenika telefonskih brojeva. U imenik je moguće dodati novi broj, obrisati postojeći i odštampati sve brojeve imenika.

Zadatak 3 rešenje:



```
imenik.py - C:/Users/Dule/Desktop/imenik.py
File Edit Format Run Options Windows Help
class Imenik:
    listaBrojeva = []
    def dodajBroj(self, broj):
        self.listaBrojeva.append(broj)
    def stampajListu(self):
        print(self.listaBrojeva)
    def obrisiBroj(self, broj):
        if broj in self.listaBrojeva:
            self.listaBrojeva.remove(broj)
        else:
            print("Poslati broj ne postoji u imeniku")
```

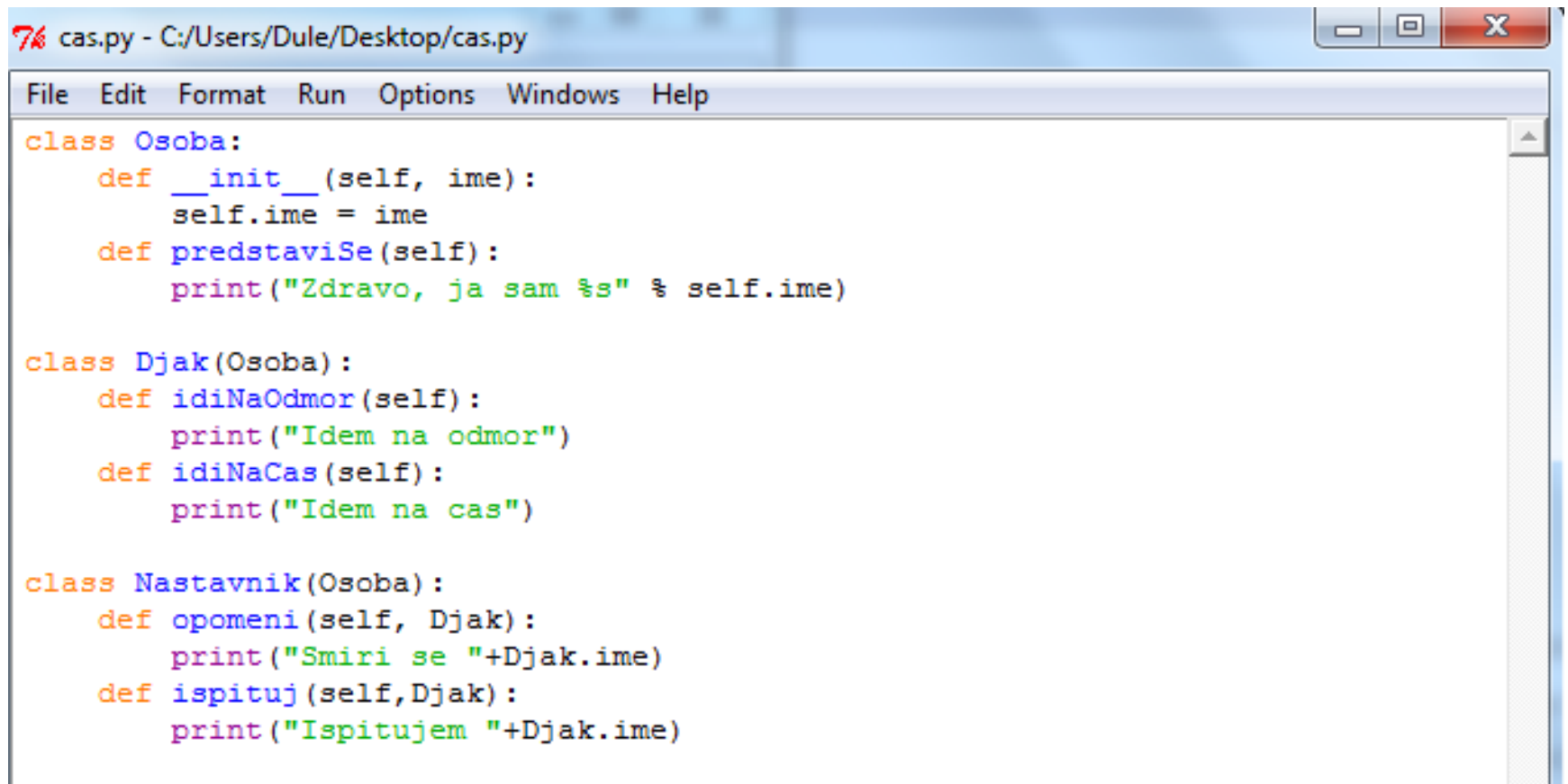
Zadatak 3 rešenje:

```
Python 3.3.3 Shell
File Edit Shell Debug Options Windows Help
Python 3.3.3 (v3.3.3:c3896275c0f6, Nov 18 2013, 21:18:40) [MSC v.1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
>>> imenik = Imenik()
>>> imenik.dodajBroj(123123)
>>> imenik.dodajBroj(456456)
>>> imenik.stampajListu()
[123123, 456456]
>>> imenik.obrisiBroj(123123)
>>> imenik.stampajListu()
[456456]
>>> imenik.obrisiBroj(123123)
Poslati broj ne postoji u imeniku
>>>
```

Zadatak 4:

- Novi predmet počinje sa nastavom. Ovaj predmet predaje jedan nastavnik a sluša ga dva djaka. Na prvom času se prvo svi predstave jedni drugima. Nastavnik može da ispituje jednog djaka ili da opomene djaka ukoliko je nemiran. Djak može da ide na odmor i da ide na čas.

Zadatak 4:



```
cas.py - C:/Users/Dule/Desktop/cas.py
File Edit Format Run Options Windows Help
class Osoba:
    def __init__(self, ime):
        self.ime = ime
    def predstaviSe(self):
        print("Zdravo, ja sam %s" % self.ime)

class Djak(Osoba):
    def idiNaOdmor(self):
        print("Idem na odmor")
    def idiNaCas(self):
        print("Idem na cas")

class Nastavnik(Osoba):
    def opomeni(self, Djak):
        print("Smiri se "+Djak.ime)
    def ispituj(self, Djak):
        print("Ispitujem "+Djak.ime)
```

Zadatak 4:

```
Python 3.3.3 Shell
File Edit Shell Debug Options Windows Help
Python 3.3.3 (v3.3.3:c3896275c0f6, Nov 18 2013, 21:18:40) [MSC v.1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
>>> nastavnik = Nastavnik("Pera")
>>> djak1 = Djak("Milos")
>>> djak2 = Djak("Jelena")
>>> djak1.idiNaCas()
Idem na cas
>>> djak2.idiNaCas()
Idem na cas
>>> nastavnik.predstaviSe()
Zdravo, ja sam Pera
>>> djak1.predstaviSe()
Zdravo, ja sam Milos
>>> djak2.predstaviSe()
Zdravo, ja sam Jelena
>>> nastavnik.opomeni(djak1)
Smiri se Milos
>>> nastavnik.ispituj(djak2)
Ispitujem Jelena
>>> djak1.idiNaOdmor()
Idem na odmor
>>> djak2.idiNaOdmor()
Idem na odmor
>>>
```