

Programiranje za umetnike 2

~ Objektno orijentisano programiranje ~

Staša Vujičić Stanković

Teme

Objektno orijentisano
programiranje

-

osnovni koncepti

Objektno orijentisano programiranje

- Kada se koristi programski jezik koji nije OO, rešenja se izražavaju u terminima brojeva i karaktera.
- Kod OO programskih jezika na brojeve i karaktere se gleda kao na primitivne tipove, a problemi se rešavaju u terminima entiteta, tj. objekata koji se javljaju u kontekstu problema.
- Python je objektno orijentisan jezik.

Objektno orijentisano programiranje

- Problem: Organizacija voznog parka u firmi.
- Šta su objekti?
- Neki od objekata:
prevozno sredstvo (generalno posmatrano),
ali i autobus, automobil...

Objektno orijentisano programiranje

- **Objekat** – integralna celina podataka i procedura za rad sa njima.
- **OOP** – programska paradigma zasnovana na skupu objekata koji dejstvuju međusobno.
Glavne obrade se zasnivaju na manipulisanju objektima.
- U Python-u je gotovo sve objekat sa odgovarajućim skupom podataka (osobina) i metoda (funkcija) koji ga karakterišu.

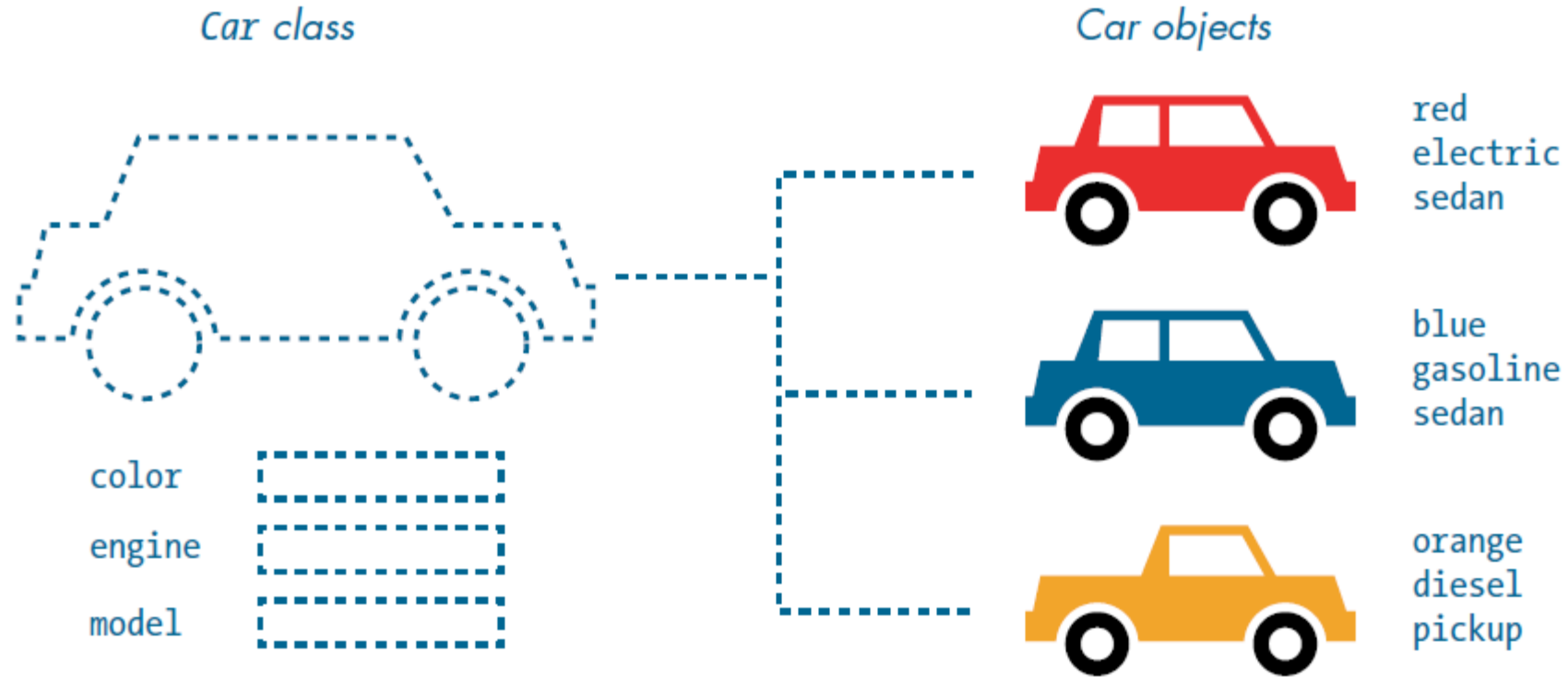
Objektno orijentisano programiranje

- **Klasa** – Skup objekata sa zajedničkim svojstvima.
Npr. prevozno sredstvo je klasa.
- **Potklasa** – klasa koja nasleđuje sve osobine roditeljske klase (**natklase**)
i ima dodatnu specijalizaciju.
Npr. automobil ima sve karakteristike klase prevozno sredstvo,
ali ima i svoje dodatne karakteristike – npr. 4 točka...

Objektno orijentisano programiranje

- **Instanca** – konkretan objekat iz klase.
- **Instancne promenljive ili atributi klase** – parametri koji definišu objekat neke klase. Mogu biti osnovni tipovi podataka ili objekti neke druge klase.
- Npr. automobil:
 - boja: bela, crvena, crna,...
 - proizvođač: Honda, Opel, BMW, ...
 - model: limuzina, sportski, karavan...

Objektno orijentisano programiranje



* preuzeto iz knjige –

Bunn, Tristan. Learn Python Visually: Creative Coding with Processing.py. No Starch Press, Inc., 2021. 8

Objektno orijentisano programiranje

- Klasa pored kolekcije podataka određuje i šta sve može da se radi sa objektima te klase. Definiše listu mogućih operacija nad objektima klase. To se postiže blokom programskog koda koji se naziva metoda.
- Na primer:
 - startuj motor
 - zaustavi auto
 - ubrzaj...

Objektno orijentisano programiranje

- Klase su grupisane u odgovarajuće skupove koji se nazivaju **moduli**.
- **Biblioteka** – kolekcija modula.
- Korisnik sam može da pravi svoje module koji će sadržati odgovarajuće klase.
- Mi ćemo u nastavku kursa koristiti postojeće biblioteke i njihove module.

Objektno orijentisano programiranje u Python-u

- Python je objektno orijentisani programski jezik.
- Skoro sve u Python-u je objekat, sa svojim atributima i metodama.
- Klasa je poput šablona za kreiranje objekata.

Kreiranje klase

- Za kreiranje klase se koristi ključna reč `class`:

```
class MojaKlasa:  
    rbr = 100
```

Kreiranje objekta klase

```
class MojaKlasa:
```

```
    rbr = 100
```

```
objekat = MojaKlasa()
```

```
print(objekat.rbr)
```

```
class NazivKlase:
```

```
    <konstruktor>
```

```
    <metodi klase>
```

__init__() funkcija

- Sve klase imaju __init__() funkciju, koja se uvek izvršava kada se klasa inicijalizuje.
- Funkcija __init__() se koristi da bi se dodelile vrednosti atributima objekta ili drugim operacijama koje su neophodne kada se objekat kreira.
- Funkcija __init__() se poziva automatski svaki put kada se klasa koristi za kreiranje novog objekta.

Konstruktor

- Nakon definisanja klase, objekti se kreiraju pomoću konstruktora, koji:
 - pokreće funkciju `__init__` za inicijalizaciju objekta
 - kreira u memoriji objekat određene klase
- Sintaksa konstruktora `NazivKlase (argumenti)`

Pristup članovima objekata

- Članovi objekata su atributi i metodi
- Pristup članovima se vrši pomoću naziva objekta i operatora „.“:

`NazivObjekta = NazivKlase(argumenti)`

`NazivObjekta.atribut`

`NazivObjekta.metod(argumenti)`

Pristup članovima objekata

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

```
osoba1 = Osoba("Jovan", "Jovanovic", 20)
```

```
print(osoba1.ime)  
print(osoba1.prezime)  
print(osoba1.starost)
```

Predefinisane vrednosti

- Atributima se mogu dodeliti predefinisane vrednosti.
- Na primer,

```
class Osoba:
```

```
    def __init__(param, ime, prezime, starost):  
        param.ime = ime  
        param.prezime = prezime  
        param.starost = starost  
        param.zanimanje = "STUDENT"
```

Metode

- Objekti takođe mogu da sadrže metode.
- Metode u objektima su funkcije koje pripadaju objektu.
- Sintaksa za pokretanje metoda nekog objekta je:

`NazivObjekta.metod(argumenti)`

Objektno orijentisano programiranje

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

    def jasm(self):
        print("Ja sam " + self.ime+ " " +self.prezime + ".")

osoba1 = Osoba("Jovan", "Jovanovic", 20)
osoba1.jasm()
```

self parametar

- Parametar `self` je referenca na trenutnu instancu klase i koristi se za pristup promenljivim koje pripadaju klasi.
- Metodi npr. unutar objekta koriste kao prvi argument `self`, referencu na sam objekt čija vrednost se definiše prilikom kreiranja objekta

Postavljanje vrednosti atributa objekata

- Vrednosti atributa se postavljaju tačka notacijom.

- Na primer,

```
osoba1.ime = "Aleksandar"
```

- Atribut se briše upotrebom ključne reči del.

```
del osoba1.ime
```

Metode

- U Python-u postoje tri tipa metoda, a to su instancni, klasni i statički.
 - Instancni uzima `self` kao prvi argument. Nazivaju se još i objektni ili regularni metodi.
 - Klasni metod uzima `cls` kao prvi argument. `cls` se odnosi na klasu. Da bismo pristupili promenljivoj klase unutar metode, koristimo tzv. dekorator `@classmethod` i prosleđujemo klasu metodu.
 - Statički metod ne uzima ništa kao prvi argument. Ima ograničenu upotrebu.

Literatura

- Lutz, Mark. Learning Python: Powerful Object-Oriented Programming. O'Reilly Media, Inc., 2013.
- [Wentworth, Peter, Elkner, Jeffrey, Downey, Allen B. and Meyers, Chris. How to Think Like a Computer Scientist: Learning with Python 3. free online book](#)
- Bunn, Tristan. Learn Python Visually: Creative Coding with Processing.py. No Starch Press, Inc., 2021.
- ...

Hvala



Stasa Vujičić Stanković



stasa@math.rs



www.matf.bg.ac.rs/~stasa