

Uvod v programiranje

6. vaje

Kenny Štorgel

Fakulteta za informacijske študije Novo mesto

Novo mesto, februar 2019

Objektno orientirano programiranje (OOP)

- ▶ Eden izmed pristopov k programiranju
- ▶ Pri funkcijskem programiranju so enote funkcije, pri objektnem pa razredi (`class`)
- ▶ Modularnost in večkratna uporabnost rešitve

- ▶ Osnovne enote pri izvajanju programov
- ▶ Imajo definirane lastnosti - stanja in funkcionalnosti - metode, s katerimi spreminjajo stanja oziroma objekt
- ▶ Stanja se shranjujejo v **polja**, to so spremenljivke definirane v razredu, vendar izven vseh metod

- ▶ Definicija oziroma načrt objekta je shranjen v razredu (**class**)
- ▶ Posamezen objekt je primerek (**instance**) razreda
- ▶ En razred **Clovek** lahko vsebuje večino opisov lastnosti in funkcionalnosti človeka, primerkov pa je nekaj milijard...
- ▶ Primer: razred **Clovek** ima npr. stanja teža, starost, barva las ter funkcionalnost shujšaj.

- ▶ **Naloga 1:** V razred `Clovek` dodajte polje, ki določa višino človeka. V neprazen konstruktor dodajte določitev te lastnosti. Ustvarite še metodo `zrasti`, ki bo kot parameter prejela količino, za katero se naj spremeni višina danega objekta. Popravite tudi izpis.
- ▶ **Pozor!** Če ima parameter konstruktorja enako ime kot polje, potem morate pri prirejanju vrednosti uporabiti `this`, da prevajalnik ve, da prirejate vrednost globalni spremenljivki!

- ▶ Vrednosti lastnosti objektov bomo določali z metodami tipa `setLastnost` in pridobivali z metodami tipa `getLastnost`
- ▶ Primer je v datoteki `Avto.java` v paketu nalog!

- ▶ **Naloga 2:** V razred `Avto` dodajte polje, ki določa maksimalno hitrost in dodajte konstruktor, ki jo prejme, v ostalih nastavite privzeto vrednost. Nato popravite metodo `pospesi` tako, da bo pri prekoračenju maksimalne hitrosti hitrost postavila na maksimalno.

Maksimalna hitrost

- ▶ **Naloga 3:** V razred `Avto` dodajte še metodo tipa `get`, ki vrne maksimalno hitrost, popravite izpis in napišite metodo tipa `set`, ki bo maksimalno hitrost avta spremenila.

Vrste spremenljivk

- ▶ **Spremenljivke razreda:**
 - ▶ polja, ki imajo še lastnost `static`
 - ▶ samo ena za razred
 - ▶ če so tudi `final`, se ne morejo spremeniti (so `konstante`)
- ▶ **Spremenljivke objekta:**
 - ▶ polja, ki nimajo lastnosti `static`
 - ▶ vsak objekt ima svojo kopijo
- ▶ **Lokalne spremenljivke:**
 - ▶ spremenljivke definirane znotraj metode
- ▶ **Parametri:**
 - ▶ spremenljivke podane metodi

Naloga 4: Recimo, da imamo tri košare s krompirjem. Krompir je lahko bel ali rdeč. Definirajte razred `Kosara`, ki ima lastnosti `stBelihKrompirjev` in `stRdecihKrompirjev`, dva konstruktorja - praznega in konstruktor, ki kot parameter dobi vrednosti obeh polj, ter metodo `prestaviKrompir`, ki kot parameter dobi celo število `barva` tipa `byte` ter nek objekt `druga` tipa `Kosara`. Če je `barva` enaka 0, se iz košare `druga` prestavi en rdeč krompir v košaro, katere metodo smo klicali, sicer se prestavi bel krompir.

Naloga 5: Ustvarite tri košare, v prvi imejte 5 belih in 0 rdečih krompirjev, v drugi 3 bele in 12 rdečih, v tretji pa 4 bele in 10 rdečih. S pomočjo metode `prestaviKrompir` preložite vse rdeče krompirje iz druge in tretje košare v prvo, nato pa vse bele iz prve in druge v tretjo. Izdelajte še metodo `izpis`, ki izpiše koliko krompirjev je v košari ter jo pokličite z vsako izmed košar.

6. domača naloga

- ▶ Za razred `Avto` si izmislite še dve lastnosti avta, zanju naredite polji, dodajte ju v konstruktor, določite privzeti vrednosti in napišite metodi za upravljanje z njima. Popravite tudi izpis.