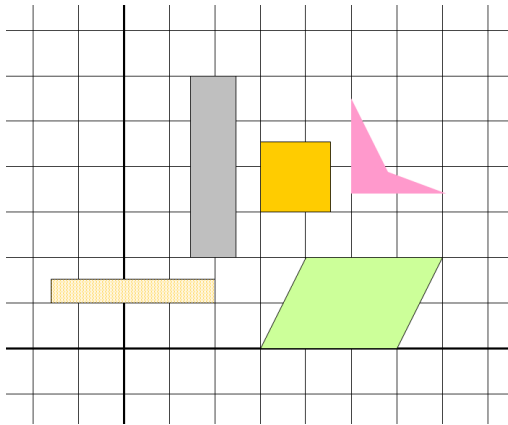


7.11.1 Metode in spremenljivke v izpeljanih razredih

Osnovni postopek pri dedovanju je izdelava novih razredov iz že obstoječih. Pri tem se razred, iz katerega izpeljujemo nove razrede, imenuje **nadrazred** (angl. *superclass*), novonastali razredi pa **izpeljani razredi** ali **podrazredi**. Tudi iz izpeljanega razreda lahko izpeljemo nove razrede. Nato iz nastalih razredov ustvarjamo **objekte**.



Primer: V naslednji nalogi načrtujemo delo z velikim številom štirikotnih likov. Predstavljamo si jih v nekem koordinatnem sistemu, so različnih vrst in so različno obarvani.

Za vse štirikotnike imamo torej

- po dva podatka o položaju v koordinatnem sistemu (recimo koordinati težišča),
- podatek o barvi.

Slika 54: Štirikotniki različnih vrst. Skupne lastnosti so: dve koordinati položaja in barva

Po gornjem opisu napišimo razred *Štirikotnik* za izdelavo "splošnih" štirikotnikov:

```
public class Štirikotnik{
    int x, y; // koordinati položaja
    String barva;

    // konstruktor
    public Štirikotnik(int iks, int ipsilon, String ba){
        x = iks;
        y = ipsilon;
        barva = ba; }
}
```

Z zapisanim konstruktorjem lahko izdelamo "štirikotnik" z določenim položajem v koordinatnem sistemu (določijo se vrednosti koordinat x in y) in barvo. Vendar ni določeno, ali gre za pravokotnik, kvadrat, paralelogram ali kateri drug štirikotnik. Zato tudi nimamo spremenljivk, značilnih za posamezno vrsto štirikotnika. Razred *Štirikotnik* naj zaenkrat služi kot osnova za izdelavo bolj določenih štirikotnikov.

Podatki o stranicah pa so pri različnih štirikotnikih različni:

- za pravokotnik potrebujemo dva podatka, dolžini dveh stranic,
- kvadrat ima vse stranice enake, zadošča samo en podatek (dolžina stranice),
- nepravilni štirikotnik ima lahko vse štiri stranice različno dolge ...

Za različne like moramo podatke o stranicah posredovati različno. Kasneje bomo tudi njihove ploščine računali po različnih formulah.

Iz razreda *Stirikotnik* izpeljimo nov razred, *Pravokotnik*, ki od razreda *Stirikotnik* prevzame spremenljivke *x*, *y* in *barva*, **dodamo** mu pa **spremenljivki** za dolžini stranic ter **metodo** za izpis ploščine. Dejstvo, da razred *Pravokotnik* izpeljujemo iz razreda *Stirikotnik*, povemo z rezervirano besedo *extends*, ki v deklaraciji metode "pokaže" na prednika, nadrazred:

```
public class Pravokotnik extends Stirikotnik{
    int a, b;
    //konstruktorji
    //metoda za ploščino
    //ostalo
}
```

Razred *Pravokotnik* smo izpeljali iz razreda *Stirikotnik*. Razred *Stirikotnik* je nadrazred razreda *Stirikotnik*.

7.11.2 Konstruktorji v izpeljanih razredih

Za ustvarjanje "pravokotnikov" si izdelamo konstruktor tudi v razredu *Pravokotnik*, saj konstruktor nadrazreda *Stirikotnik* ne zadošča – ne zna uporabiti stranic pravokotnika. Konstruktor za nove pravokotnike naj omogoča določitev začetnih vrednosti vseh spremenljivk, npr. takole:

```
Pravokotnik pr1 = new Pravokotnik(12,4,"bela",3,4);
```

Prva dva podatka sta koordinati položaja, sledi barva in nato dolžini stranic pravokotnika.

Vidimo, da prvi trije podatki izvirajo še iz nadrazreda *Stirikotnik*, zadnja dva (dolžini stranic) pa iz izpeljanega razreda *Pravokotnik*. To se bo odražalo tudi v konstruktorju. V njem pravzaprav najprej pokličemo konstruktor nadrazreda in mu posredujemo podatke, ki izvirajo iz nadrazreda (koordinati položaja *x* in *y* ter *barva*). Nove spremenljivke izpeljanega razreda pa vgradimo kot običajno:

```
public Pravokotnik(int xx,int yy,String ba,int aa,int bb){
    super(xx, yy, ba); // kličemo konstruktor nadrazreda
    this.a = aa;
    this.b = bb;
}
```

Konstruktor nadrazreda smo poklicali z besedico *super*, ki je v Javi rezervirana beseda.

V izpeljanem razredu, razredu *Pravokotnik*, dodajmo še metodo *izpisPloscine()*. To bo objektna metoda, ki iz dolžin stranic pravokotnika izračuna in izpiše njegovo ploščino. Recimo takole:

```
public void izpisPloscine(){
    System.out.println("Ploscina pravokotnika je " + (this.a * this.b));
}
```

Celoten zapis izpeljanega razreda *Pravokotnik* je takšen:

```
public class Pravokotnik extends Stirikotnik{
    int a,b; // dolžine stranic

    public Pravokotnik(int xx,int yy,String bar,int aa,int bb){
```