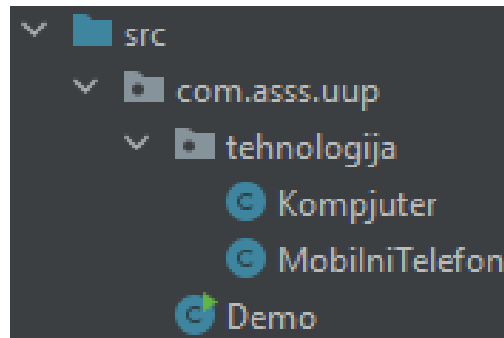


VEŽBA 9: OOP – Klase i objekti

Primer 1 – Paketi, klase i objekti, konstruktori, pristupni i metodi mutatori, ključne reči „static“ i „this“, prenos po vrednosti i referenci



```
package com.asss.uup.tehnologija;

public class Kompjuter {

    // polja klase Kompjuter - svojstva objekata (instanci) klase Kompjuter
    private String maticnaPloca;
    private String procesor;
    private int hardDisk;
    private int ram;
    private String grafickaKartica;

    static private String operativniSistem = "Windows 10";
```

```
// konstruktori klase Kompjuter

/*
    prazan konstruktor
    (ovo je oblik podrazumevanog konstrukora koji je moguće koristiti
    samo kada nema ni jednog definisanog u bilo kom obliku)
*/
public Kompjuter() {
}

// konstruktor koji prima samo dva argumenta i dodeljuje vrednost istoimenim poljima
public Kompjuter(String maticnaPloca, String procesor) {
    this.maticnaPloca = maticnaPloca;
    this.procesor = procesor;
}

// konstruktor koji prima cetiri argumenta i koji implicitno poziva prethodni konstruktor (pomocu this)
public Kompjuter(String maticnaPloca, String procesor, int hardDisk, int ram) {
    this(maticnaPloca, procesor);
    this.hardDisk = hardDisk;
    this.ram = ram;
}

// konstruktor koji prima dodeljuje vrednost svim poljima objekta
public Kompjuter(String maticnaPloca, String procesor, int hardDisk, int ram, String grafickaKartica) {
    this.maticnaPloca = maticnaPloca;
    this.procesor = procesor;
    this.hardDisk = hardDisk;
    this.ram = ram;
    this.grafickaKartica = grafickaKartica;
}

/*
    konstruktor koji treba da omoguci pravljenje istog
    objekta kao sto je onaj koji je predat kao argument
*/
public Kompjuter(Kompjuter kompjuter) {
    this.maticnaPloca = kompjuter.getMaticnaPloca();
    this.procesor = kompjuter.getProcesor();
    this.hardDisk = kompjuter.getHardDisk();
    this.ram = kompjuter.getRam();
    this.grafickaKartica = kompjuter.getGrafickaKartica();
}
```

```
// pristupni metodi (getters) i metodi mutatori (setters) za sva polja klase
public String getMaticnaPloca() { return maticnaPloca; }

public void setMaticnaPloca(String maticnaPloca) { this.maticnaPloca = maticnaPloca; }

public String getProcesor() { return procesor; }

public void setProcesor(String procesor) { this.procesor = procesor; }

public int getHardDisk() { return hardDisk; }

public void setHardDisk(int hardDisk) { this.hardDisk = hardDisk; }

public int getRam() { return ram; }

public void setRam(int ram) { this.ram = ram; }

public String getGrafickaKartica() { return grafickaKartica; }

public void setGrafickaKartica(String grafickaKartica) { this.grafickaKartica = grafickaKartica; }

// generisani toString() metod za prikaz objekata (u suprotnom se prikazuje hes kod objekta)
@Override
public String toString() {
    return "Kompjuter{" +
        "maticnaPloca='" + maticnaPloca + '\'' +
        ", procesor='" + procesor + '\'' +
        ", hardDisk=" + hardDisk +
        ", ram=" + ram +
        ", grafickaKartica='" + grafickaKartica + '\'' +
        '}';
}
```

```
package com.asss.uup.tehnologija;

public class MobilniTelefon {

    // polja klase MobilniTelefon - svojstva objekata (instanci) klase MobilniTelefon
    private String proizvođjac;
    private int dijagonala;

    public MobilniTelefon() {
    }

    public MobilniTelefon(String proizvođjac, int dijagonala) {
        this.proizvođjac = proizvođjac;
        this.dijagonala = dijagonala;
    }

    public String getProizvođjac() { return proizvođjac; }

    public void setProizvođjac(String proizvođjac) { this.proizvođjac = proizvođjac; }

    public int getDijagonala() { return dijagonala; }

    public void setDijagonala(int dijagonala) { this.dijagonala = dijagonala; }

    @Override
    public String toString() {
        return "MobilniTelefon{" +
            "ime='" + proizvođjac + '\\\'' +
            ", dijagonala=" + dijagonala +
            '}';
    }
}
```

```
package com.asss.uup;

import com.asss.uup.tehnologija.Kompjuter;

public class Demo {

    public static void main(String[] args) {

        // pravljenje dve instance tipa Kompjuter
        Kompjuter komp1 = new Kompjuter();
        Kompjuter komp2 = new Kompjuter( maticnaPloca: "ROG Maximus", procesor: "Intel i9", hardDisk: 2, ram: 64, grafickaKartica: "GeForce RTX");

        // postavljanje vrednosti dva svojstva prve instance
        komp1.setMaticnaPloca("ROG Hero");
        komp1.setProcesor("Intel i7");

        // prikaz instance
        System.out.println(komp1);
        System.out.println(komp2.toString());

        System.out.println("\n-----\n");
```

```
System.out.println("\n-----\n");

// pravljenje nova instance tipa Kompjuter
Kompjuter desk = new Kompjuter( maticnaPloca: "ROG Strix", procesor: "Intel i9", hardDisk: 3, ram: 64, grafickaKartica: "GeForce RTX");

// dodela novonapravljenog objekta novoj promenljivoj (dodeljuje se SAMO REFERENCA NA NJEGA)
Kompjuter laptop = desk;

// pravljenje novog objekta od postojećeg
// Kompjuter laptop = new Kompjuter(desk);

// prikaz objekta/objekata u promenljivama desk i laptop
System.out.println(desk);
System.out.println(laptop);

// pravljenje dva podatka prostih tipova
int n = 7;
int s = n;

// prikaz podataka prostih tipova
System.out.println(n);
System.out.println(s);

System.out.println("\n-----\n");
```

```
System.out.println("\n-----\n");

// promena vrednosti svojstava objekta koji se cuva u promenljivoj laptop
laptop.setProcesor("Intel i7");
laptop.setGrafickaKartica("GeForce GTX");
laptop.setHardDisk(1);
laptop.setRam(32);

// prikaz objekta/objekata u promenljivama desk i laptop nakon promene svojstava
System.out.println(desk);
System.out.println(laptop);

// promena vrednosti podatka prostog tipa koji se cuva u promenljivoj s
s = 12;

// prikaz podataka prostih tipova nakon promene
System.out.println(n);
System.out.println(s);
}
```

```
Kompjuter{maticnaPloca='R0G Hero', procesor='Intel i7', hardDisk=0, ram=0, grafickaKartica='null'}
Kompjuter{maticnaPloca='R0G Maximus', procesor='Intel i9', hardDisk=2, ram=64, grafickaKartica='GeForce RTX'}

-----

Kompjuter{maticnaPloca='R0G Strix', procesor='Intel i9', hardDisk=3, ram=64, grafickaKartica='GeForce RTX'}
Kompjuter{maticnaPloca='R0G Strix', procesor='Intel i9', hardDisk=3, ram=64, grafickaKartica='GeForce RTX'}
7
7

-----

Kompjuter{maticnaPloca='R0G Strix', procesor='Intel i7', hardDisk=1, ram=32, grafickaKartica='GeForce GTX'}
Kompjuter{maticnaPloca='R0G Strix', procesor='Intel i7', hardDisk=1, ram=32, grafickaKartica='GeForce GTX'}
7
12
```