

Laboratorijske vežbe – dvočas br. 7

Zadatak 1

(radi ga laborant u saradnji sa studentima)

Napraviti klasu **KucniAparat** koja ima:

- Atribut ukljucen koji ima vrednost TRUE ako je aparat uključen, a u suprotnom FALSE.
- Konstruktor koji kao ulazni argument prima boolean vrednost i postavlja je kao vrednost atributa ukljucen.
- Metodu ukljuci koja ukljucuje aparat tj. postavlja vrednost atributa ukljucen na TRUE.
- Metodu iskljuci koja iskljucuje aparat tj. postavlja vrednost atributa ukljucen na FALSE.
- Metodu ispisi koja ispisuje na ekranu poruku o tome da li je aparat ukljucen ili nije.

Napraviti klasu **Televizor** koja nasleđuje klasu KucniAparat i ima (*ključna reč EXTENDS*):

- Atribut program koji predstavlja broj programa koji se trenutno prikazuje.
- Konstruktor koji kao ulazni argument prima boolean vrednost i postavlja je kao vrednost atributa ukljucen. Ovaj konstruktor ima i još jedan ulazni argument koji predstavlja početni broj programa. Ako je uneta vrednost manja od jedan ili veća od 40, ispisuje se greška na ekranu a atribut program dobija vrednost 1. U suprotnom, atribut program dobija unetu vrednost. (*ključna reč SUPER*)
- Metodu promeniProgramNavise koja menja program na sledeći. Ako se na televizoru već pušta 40. program, program dobija vrednost 1.
- Metodu promeniProgramNanize koja menja program na prethodni. Ako se na televizoru već pušta 1. program, program dobija vrednost 40.
- Metodu ispisi koja, pored poruke o tome da li je uključen, ispisuje i poruku o tome koji program se trenutno prikazuje. (*override - redefinisane metoda*)

(Ovu klasu studenti rade sami)

Napraviti klasu **Radio** koja nasleđuje klasu KucniAparat i ima:

- Atribut frekvencija koji predstavlja trenutnu frekvenciju koju pušta radio aparat (npr. 87,5).
- Konstruktor koji kao ulazni argument prima boolean vrednost i postavlja je kao vrednost atributa ukljucen. Ovaj konstruktor ima i još jedan ulazni argument koji predstavlja početnu frekvenciju radio aparata. Ako je uneta frekvencija van opsega 87.5 – 107.9 ispisuje se poruka o grešci na ekranu, a atribut frekvencija dobija vrednost 87.5. U suprotnom, atribut frekvencija dobija unetu vrednost.
- Metodu getFrekvencija koja vraća trenutnu vrednost atributa frekvencija.
- Metodu setFrekvencija koja kao ulazni argument prima novu frekvenciju. Ako je uneta frekvencija van opsega 87.5 – 107.9 ispisuje se poruka o grešci na ekranu. U suprotnom, atribut frekvencija dobija unetu vrednost.
- Metodu ispisi koja ispisuje samo poruku o frekvenciji koju radio pušta.

Napraviti klasu **TestKucniAparat** koja kreira po jedan objekat klase KucniAparat, Radio i Televizor. I poziva neke njihove metode. (*ovde objasniti kako funkcionišu redefinisane metode*)

```
public class KucniAparat {  
  
    boolean ukljucen;  
  
    public void ukljuci () {  
        ukljucen = true;  
    }  
  
    public void iskljuci () {  
        ukljucen = false;  
    }  
}
```

```

    }

    public KucniAparat (boolean uk){
        ukljucen = uk;
    }

    public void ispisi (){
        if (ukljucen) System.out.println("Aparat je ukljucen");
        else System.out.println("Aparat je iskljucen");
    }
}

public class Televizor extends KucniAparat {

    int program;

    public Televizor (boolean uk, int p){
        super(uk);

        if ((p < 1) || (p > 40)) {
            System.out.println("Greska");
            program = 1;
        }
        else program = p;
    }

    public void promeniProgramNavise(){
        if (program < 40) program++;
        else program = 1;
    }

    public void promeniProgramNanize(){
        if (program > 1) program--;
        else program = 40;
    }

    public void ispisi (){
        super.ispisi();
        System.out.println("Trenutno je pusten "+program+". program");
    }
}

public class Radio extends KucniAparat {

    double frekvencija;

    public Radio (boolean uk, double f){
        super(uk);

        if ((f < 87.5) || (f > 107.9)) {
            System.out.println("Greska");
            frekvencija = 87.5;
        }
        else frekvencija = f;
    }

    public double getFrekvencija() {
        return frekvencija;
    }

    public void setFrekvencija(double frekvencija) {
        if ((frekvencija >= 87.5) && (frekvencija <= 107.9))
            this.frekvencija = frekvencija;
    }
}

```

```

        else
            System.out.println("Greska");
    }

    public void ispisi () {
        System.out.println("Frekvencija je "+frekvencija+" Mhz");
    }
}

public class TestKucniAparat {

    public static void main(String[] args) {

        KucniAparat k = new KucniAparat(true);

        k.iskljuci();

        System.out.println("Ispisivanje podataka o kucnom aparatu");
        k.ispisi();

        Televizor t = new Televizor (true, 33);

        t.iskljuci();
        t.promeniProgramNanize();

        System.out.println("Ispisivanje podataka o televizoru");
        t.ispisi();

        Radio r = new Radio (false, 99.5);

        r.ukljuci();
        r.setFrekvencija(88.9);

        System.out.println("Ispisivanje podataka o radiju");
        r.ispisi();

    }

}

```

Zadatak 2

(radi ga laborant u saradnji sa studentima)

Napraviti apstraktnu klasu **Zaposleni** koja ima (*ključna reč ABSTRACT i apstraktne klase*):

- Atribut plata koji je realan broj.
- Metodu getPlata koja vraća trenutnu vrednost atributa plata.
- Apstraktnu metodu izracunajPlatu koja ne vraća ništa, ali ima ulazni argument koji predstavlja broj sati koji je zaposleni radio. (*apstraktne metode*)

Napraviti klasu **KancelarijskiRadnik** koja nasleđuje klasu Zaposleni i:

- Realizuje metodu izracunajPlatu. Ova metoda postavlja vrednost atributa plata prema formuli $PLATA = 100 * BROJ_SATI$. (*implemetacija apstraktne metode*)

Napraviti klasu **Menadzer** koja nasleđuje klasu Zaposleni i:

- Realizuje metodu izracunajPlatu. Ova metoda postavlja vrednost atributa plata prema formuli $PLATA = 1000 * BROJ_SATI$.

Napraviti klasu **TestZaposleni** i kreirati po jedan objekat klase KancelarijskiRadnik i Menadzer.

Izracunati i ispisati plate obojice ako se zna da je Menadzer radio 250 sati a kancelarijski radnik 200 sati.

```
public abstract class Zaposleni {

    double plata;

    public double getPlata() {
        return plata;
    }

    public abstract void izracunajPlatu (int broj_sati);
}

public class KancelarijskiRadnik extends Zaposleni {

    public void izracunajPlatu(int broj_sati) {
        plata = 100 * broj_sati;
    }

}

public class Menadzer extends Zaposleni {

    public void izracunajPlatu(int broj_sati) {
        plata = 1000 * broj_sati;
    }

}

public class TestZaposleni {

    public static void main(String[] args) {

        KancelarijskiRadnik k = new KancelarijskiRadnik();
        Menadzer m = new Menadzer();

        m.izracunajPlatu(250);
        k.izracunajPlatu(200);

        System.out.println("Menadzerova plata je: "+m.getPlata());
        System.out.println("Plata kancelarijskog radnika je: "+
            k.getPlata());
    }

}
```