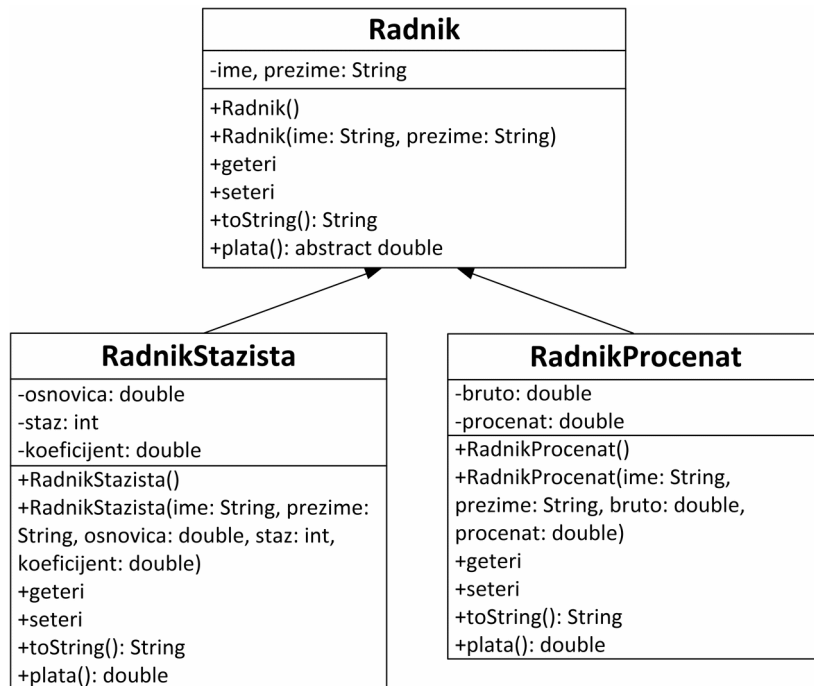


Vježbe 6



Slika 1. Klasa Radnik i izvedene klase RadnikStazista i RadnikProcenat

Na slici 1. su prikazane klasa Radnik i izvedene klase RadnikStazista i RadnikProcenat. Realizovati ove tri klase u potpunosti, pri čemu su svi podaci članovi i metode prikazani na slici. Oznake – i + ispred podataka i metoda označavaju privatno i javno pravo pristupa, respektivno. Metoda plata u klasi Radnik je apstraktna.

Plata radnika stažiste se računa kao:

$$\text{plata} = \text{osnovica} * \text{koeficijent} * (1 + \text{staz}/50),$$

dok se plata radnika na procenat računa kao:

$$\text{plata} = \text{procenat} * \text{bruto}.$$

Pri pokušaju unosa negativne osnovice, staža ili koeficijenta, baciti izuzetak tipa `IllegalArgumentException`.

Kreirati klasu `RadnikTest` koja će testirati prethodno kreirane klase. Testiranje klasa podrazumijeva kreiranje bar po 3 instance klasa `RadnikStazista` i `RadnikProcenat`, i pozivanje realizovanih metoda.

Konačno, reference na sve kreirane objekte smjestiti u jedan niz, proći kroz taj niz, odrediti ukupan iznos plata radnika stažista i radnika na procenat. Odštampati dobijene iznose.

```

public abstract class Radnik {
    String ime;
    String prezime;
    Radnik(){
        setIme(null);
        setPrezime(null);
    }

    Radnik(String a, String b){
        setIme(a);
        setPrezime(b);
    }

    public void setIme(String a) {
        ime = a;
    }

    public void setPrezime(String a) {
        prezime = a;
    }

    public String getIme() {
        return ime;
    }

    public String getPrezime() {
        return prezime;
    }

    @Override
    public String toString() {
        return "Radnik [ime=" + getIme() + ", prezime=" + getPrezime() + "];"
    }

    public abstract double plata();
}

```

```

public class RadnikStazista extends Radnik {
    double osnovica;
    int staz;
    double koeficijent;

    RadnikStazista(){
//    super(null, null);
        setOsnovica(0);
        setStaz(0);
        setKoeficijent(0);
    }

    RadnikStazista(String a, String b, double c, int d, double e){
        super(a, b);
        setOsnovica(c);
        setStaz(d);
        setKoeficijent(e);
    }
}

```

```

public double getOsnovica() {
    return osnovica;
}

public void setOsnovica(double a) throws IllegalArgumentException{
    if(a < 0) {
        throw new IllegalArgumentException("Osnovica mora biti > 0");
    }else {
        osnovica = a;
    }
}

public int getStaz() {
    return staz;
}

public void setStaz(int a) throws IllegalArgumentException{
    if(a < 0) {
        throw new IllegalArgumentException("Staz mora biti > 0");
    }else {
        staz = a;
    }
}

public double getKoeficijent() {
    return koeficijent;
}

public void setKoeficijent(double a) throws IllegalArgumentException{
    if(a < 0) {
        throw new IllegalArgumentException("Koeficijent mora biti > 0");
    }else {
        koeficijent = a;
    }
}

@Override
public String toString() {
    return "RadnikStazista [
        osnovica=" + getOsnovica() +
        ", staz=" + getStaz() +
        ", koeficijent=" + getKoeficijent() +
        ", ime=" + getIme() +
        ", prezime=" + getPrezime() + "];
}

public double plata() {
    return getOsnovica() * getKoeficijent() * (1 + getStaz() / 50.0);
}
}

public class RadnikProcenat extends Radnik{
    double procenat;
    double bruto;

    RadnikProcenat(){
//    super(null, null);
        setProcenat(0);
        setBruto(0);
    }
}

```

```

RadnikProcenat(String a, String b, double c, double d){
    super(a, b);
    setProcenat(c);
    setBruto(d);
}

public double getProcenat() {
    return procenat;
}

public void setProcenat(double a) {
    procenat = a;
}

public double getBruto() {
    return bruto;
}

public void setBruto(double a) {
    bruto = a;
}

@Override
public String toString() {
    return "RadnikProcenat [
        procenat=" + getProcenat() +
        ", bruto=" + getBruto() +
        ", ime=" + getIme() +
        ", prezime=" + getPrezime() + "];"
}

public double plata() {
    return getProcenat() * getBruto();
}
}

```

```

public class RadnikTest {

    public static void main(String[] args) {

        RadnikStazista rs1 = new RadnikStazista("Marko", "Marić", 20.5, 12, 3.4);
        System.out.println(rs1);
        System.out.println("Plata " + rs1.plata());

        RadnikStazista rs2 = new RadnikStazista();
        rs2.setIme("Petar");
        rs2.setPrezime("Petrović");

        try {
            rs2.setOsnovica(-5);
        } catch (IllegalArgumentException e) {
            System.out.println(e.getMessage());
        }

        rs2.setStaz(13);
        rs2.setKoefficient(3.5);
        System.out.println(rs2);
    }
}

```

```

System.out.println("Plata " + rs2.plata());

RadnikStazista rs3 = new RadnikStazista("Ivan", "Ivanović", 21.5, 15, 5.4);
System.out.println("Ime: " + rs3.getIme());
System.out.println("Prezime: " + rs3.getPrezime());
System.out.println("Osnovica: " + rs3.getOsnovica());
System.out.println("Staž: " + rs3.getStaz());
System.out.println("Koeficijent: " + rs3.getKoeficijent());
System.out.println("Plata " + rs3.plata());

RadnikProcenat rp1 = new RadnikProcenat("Ana", "Petrović", 500, 0.30);
System.out.println(rp1);
System.out.println("Plata " + rp1.plata());

RadnikProcenat rp2 = new RadnikProcenat("Filip", "Marković", 520, 0.35);

RadnikProcenat rp3 = new RadnikProcenat("Ilija", "Filipović", 700, 0.42);

Radnik radnici[] = new Radnik[6];
radnici[0] = rs1; radnici[1] = rs2; radnici[2] = rs3;
radnici[3] = rp1; radnici[4] = rp2; radnici[5] = rp3;

double plateProcenat = 0;
double plateStazisti = 0;

for(Radnik radnik : radnici){
    if(radnik instanceof RadnikStazista) {
        plateStazisti += radnik.plata();
    }else {
        plateProcenat += radnik.plata();
    }
}
System.out.println("Plate procenat " + plateProcenat);
System.out.println("Plate stažisti " + plateStazisti);
}
}

```