

## LABORATORIJSKE VJEŽBE 4

1. Kreirati klasu Knjiga koja će za podatke članove imati: broj listova, godina izdavanja i naslov. Potrebno je da klasa, pored odgovarajućih konstruktora, posjeduje i metodu VratiGodinu() koja vraća godinu izdavanja knjige. Iz klase Knjiga izvesti klasu Udzbenik sa podacima: predmet, godina i autor i odgovarajućim konstruktorima. Klasa Udzbenik treba da posjeduje i metodu VratiGodinu() koja treba da vrati godinu na kojoj je predmet koji koristi ovaj užbenik, kao i metodu VratiGodinuKnj() koja vraća godinu izdavanja knjige. Prilikom realizacije pomenutih klasa izvršiti i preklapanje atributa. U okviru main() demonstrirati pozivanje funkcije VratiGodinu().

```
public class Knjiga {

    protected int br;
    protected int god;
    protected String naslov;

    Knjiga() {
        br = 0;
        god = 0;
        naslov = "";
    }

    Knjiga(int a, int b, String c) {
        br = a;
        god = b;
        naslov = c;
    }

    public int VratiGodinu() {
        return god;
    }

    public static void main(String[] args) {
        Udzbenik a = new Udzbenik("Biologija", 3, "P. Per", 350, 2010, "Biologija I");
        System.out.println("Poziv funkcije iz klase Udzbenik");
        System.out.println("Godina na kojoj je predmet " + a.VratiGodinu());
        System.out.println("Poziv funkcije VratiGodinuKnj()");
        System.out.println("Godina na kojoj je predmet " + a.VratiGodinuKnj());
        System.out.println("Sta mozemo nauciti ovim zadatkom?");
    }
}

class Udzbenik extends Knjiga {

    protected String predmet;
    protected int god; // ovim smo preklopili god iz klase Knjiga
    protected String autor;

    Udzbenik() {
        super();
        predmet = "";
        god = 0;
        autor = "";
    }

    Udzbenik(String a, int b, String c, int a1, int b1, String c1) {
        super(a1, b1, c1);
        predmet = a;
        god = b;
    }
}
```

```

        autor = c;
    }

    public int VratiGodinu() {
        return god;
    }

    public int VratiGodinuKnj() {
        return super.god; // sta smo ovim uradili?
    }
}

```

2. Kreirati klasu **RazlomakI** koja će za privatne podatke članove imati imenilac i brojilac. Pored odgovarajućih konstruktora potrebno je realizovati odgovarajuće metode koje će omogućiti pristup podacima članovima klase, kao i metodu koja će izvršiti konverziju razlomka u realni broj (ova metoda generiše `ArithmeticException`). U okviru klase **Razlomak** realizovati metodu **main()** kojoj se učitavaju dva broja kojima se inicijalizuje objekat klase **RazlomakI**. U okviru realizovanog koda potrebno je izvršiti obradu izuzetaka koji tom prilikom mogu nastati.

```

import java.util.Scanner;

public class Razlomak {

    public static void main(String[] args) {
        try {
            Scanner a = new Scanner(System.in);
            // Ukoliko unesete nesto sto nije cijeli broj
            // parseInt metoda ce doci izuzetak tipa NumberFormatException

            System.out.println("Unesite imenilac: ");
            int im = Integer.parseInt(a.nextLine());

            // Ukoliko unesete nesto sto nije cijeli broj
            // parseInt metoda ce doci izuzetak tipa NumberFormatException
            System.out.println("Unesite brojilac: ");
            int br = Integer.parseInt(a.nextLine());

            // Kreiranje objekta klase Razlomak i inicijalizacija argumenata
            RazlomakI r = new RazlomakI(im, br);

            // Stapanje unesenih argumenata i konverzija u realni broj
            System.out.println(r.Daj_im() + "/" + r.Daj_br() + "=" + r.Konverzija());
        } catch (NumberFormatException e) {
            System.out.println("GRESKA: Imenilac i brojilac moraju biti cijeli brojevi!"
                + e.getMessage());
        } catch (ArithmeticException e) {
            System.out.println("!!!Doslo je do pokusaja dijeljenja broja sa nulom!!!");
        } catch (Exception e) {
            System.out.println("!!!Doslo je nekog nepredvidjenog izuzetka!!! " +
                e.getMessage());
        } // Ove dvije poruke se uvijek ispisuju
        finally {
            System.out.println("Ovo se uvijek izvrsava!");
            System.out.println("Hvala na paznji!");
        }
    }
}

class RazlomakI {

    private int im, br;

```

```

Razlomak1() {
    im = 0;
    br = 0;
}

Razlomak1(int a, int b) {
    im = a;
    br = b;
}

int Daj_im() {
    return im;
}

int Daj_br() {
    return br;
}

double Konverzija() throws ArithmeticException {
    if (im == 0) { //Ukoliko je imenilac 0, dizemo izuzetak tipa
        // ArithmeticException
        ArithmeticException e= new ArithmeticException("Dijeljenje sa nulom!!!");
        throw e;
    }
    // Ukoliko je sve u redu, dijelimo imenilac i brojilac i vracamo rezultat
    return (double) im / (double) br;
}
}

```

3. Napraviti klasu **Vrijeme** sa argumentima sati, minuti i sekunde. Svi argumenti su tipa **int**. Klasa treba da ima metode **unesi** i **prikazi**. Treba napraviti i klasu **VrijemeException** u kojoj je int broj argument konstruktora klase. U glavnoj klasi treba napraviti objekat klase **Vrijeme** i pozvati njene metode **unesi** i **prikazi**. Izuzetak tipa **VrijemeException** treba da se javi ako je unijet broj sati manji od 0 ili veći od 23, broj minuta manji od 0 ili veći od 59 itd.

```

import java.io.*;
import java.util.Scanner;

public class VrijemeTest {

    public static void main(String[] args) throws IOException {
        try {
            //vr objekat klase Vrijeme koji nam služi da pozivamo metode te klase
            Vrijeme vr = new Vrijeme();
            vr.prikazi();
            //Izuzeci koji ispisuju poruke u zavisnosti od vrste greske
        } //Poruka koja se ispisuje ako unos nije u odgovarajućem rasponu
        catch (VrijemeException e) {
            System.out.println("GRESKA:" + e.getMessage());
        } //Poruka koja se ispisuje ako nismo nista unijeli
        catch (NumberFormatException e) {
            System.out.println("GRESKA:" + e.getMessage());
        } //Ostale greske
        catch (IOException e) {
            System.out.println("GRESKA:" + e.getMessage());
        }
    }
}

class Vrijeme {

```

```

Scanner input = new Scanner(System.in);
int min;
int sati;
int sec;

public Vrijeme() throws IOException, VrijemeException {

    sati = unesi("sate"); //Unos sati

//Provjeravamo da li je vrijednost unesenih sati u rasponu od 0 do 23, ako nije dize
//izuzetak VrijemeException case[0] koji ispisuje poruku za sate
    if ((sati < 0) | (sati > 23)) {
        throw new VrijemeException(0);
    }

    min = unesi("minute"); //Unos minuta

//Provjeravamo da li je vrijednost unesenih minuta u rasponu od 0 do 59
//ako nije dize izuzetak VrijemeException case[1] koji ispisuje poruku za //minute
    if ((min < 0) | (min > 59)) {
        throw new VrijemeException(1);
    }

    sec = unesi("sekunde"); //Unos sekundi

//Provjeravamo da li je vrijednost unesenih sekundi u rasponu od 0 do 59
//ako nije dize izuzetak VrijemeException case[2] koji ispisuje poruku za //sekunde
    if ((sec < 0) | (sec > 59)) {
        throw new VrijemeException(2);
    }
}

//Metoda koja stampa broj unesenih sati, minuta i sekundi
public void prikazi() {
    System.out.println("sati:" + sati + " minuta: " + min + " sekundi: "
        + sec);
}

//Metoda koja služi za unos i ispisuje koji od argumenata da unesemo
//(zamjenjuje sta sa nazivom argumenta koji treba da se unese)

int unesi(String sta) throws IOException {

    System.out.print("Unesite " + sta + " :");
    return Integer.parseInt(input.nextLine());
}

}

//Klasa koja prilikom dizanja izuzetka
//(ako se unese broj sati,minuta ili sekundi koji nije u navedenom rasponu)
//javlja odgovarajucu poruku

class VrijemeException extends Exception {
    String param;

VrijemeException(int broj) {

//U zavisnosti da li je izuzetak dignut prilikom unosa sati(case 0)
//minuta(case 1) ili sekundi(case 2)
//u param upisuje odgovarajucu poruku

    switch (broj) {

```

```
case 0 : {param = "Validan unos je broj od 0 do 23!"; break;}
case 1 : {param = "Validan unos je broj od 0 do 59!"; break;}
case 2 : {param = "Validan unos je broj od 0 do 59!"; break;}
    }
}

//Metoda koja vraca vrijednost poruka
public String getMessage() {
    return (param);
}
}
```