

Laboratorijske vežbe – dvočas br. 10 - blok nastava (studenti sami rade sve zadatke)

Zadatak 1

Napraviti javnu klasu **ReliPosada** u paketu **reli.posade** koja ima:

- Privatni atribut **vozac** koji predstavlja ime i prezime vozača dato u formatu "IME PREZIME".
- Privatni atribut **suvozac** koji predstavlja ime i prezime suvozača dato u formatu "IME PREZIME".
- Privatni atribut **vreme** koji predstavlja datum održavanja trke i vreme za koje je posada završila reli izraženo u satima, minutima i sekundama.
- Odgovarajuće javne **get** i **set** metode za ove attribute. Nedoizvoljene vrednosti za sva tri atributa su **NULL** vrednosti. U slučaju unosa nedozvoljenih vrednosti za neki atribut, potrebno je ispisati poruku o tome da ulazna vrednost za taj atribut ne može biti **NULL**.
- Redefinisano metodu **toString** klase **Object** koja vraća jedan **String** u kome se nalaze svi podaci o reli posadi u formatu "Vozač: ____ Suvozač: ____ Datum trke: ____ Vreme: ____".

Napraviti javni interfejs **IzvestavanjeOReliju** u paketu **reli.izvestavanje** koji ima:

- Javnu metodu **ispisiNajbrzePosade** koja nema ulaznih argumenata i ne vraća ništa.

Napraviti javnu klasu **ReliLista** u paketu **reli** koja implementira interfejs **IzvestavanjeOReliju** i ima:

- Privatni atribut **posade** koji predstavlja listu objekata klase **ReliPosada**. Ovu listu je potrebno odmah inicijalizovati.
- Javnu metodu **unesiUListu** koja kao ulazni argument prima objekat klase **ReliPosada** i unosi ga u listu (i to na početak) samo ako uneti objekat nema **null** vrednost. U suprotnom, potrebno je ispisati poruku koja opisuje u čemu je greška.
- Implementiranu javnu metodu **ispisiNajbrzePosade** koja na ekranu ispisuje podatke o onim posadama koje su na reliju ostvarile vreme manje od 1sata. Podatke o svakoj posadi ispisati u posebnom redu.
- Javnu metodu koja na ekranu ispisuje podatke samo o onim posadama u kojima se vozač i suvozač isto prezivaju.

```
package reli.posade;

import java.util.GregorianCalendar;

public class ReliPosada {

    private String vozac;
    private String suvozac;
    private GregorianCalendar vreme;

    public GregorianCalendar getVreme() {
        return vreme;
    }

    public void setVreme(GregorianCalendar vreme) {
        if (vreme == null)
            System.out.println("Vreme ne moze biti null");
        else
            this.vreme = vreme;
    }

    public String getVozac() {
        return vozac;
    }

    public void setVozac(String vozac) {
        if (vozac == null)
            System.out.println("Ime vozaca ne moze biti null");
        else
            this.vozac = vozac;
    }
}
```

```

    public String getSuvozac() {
        return suvozac;
    }

    public void setSuvozac(String suvozac) {
        if (suvozac == null)
            System.out.println("Ime suvozac ne moze biti null");
        else
            this.suvozac = suvozac;
    }

    public String toString(){
        //Uzimanje podataka o datumu trke
        int godina = vreme.get(GregorianCalendar.YEAR);
        int mesec = vreme.get(GregorianCalendar.MONTH)+1;
        int dan = vreme.get(GregorianCalendar.DAY_OF_MONTH);

        //Uzimanje podataka o vremenu ostvarenom na trci
        int sat = vreme.get(GregorianCalendar.HOUR);
        int minut = vreme.get(GregorianCalendar.MINUTE);
        int sekunda = vreme.get(GregorianCalendar.SECOND);

        return "Vozac: "+vozac+" Suvozac: "+suvozac+
            " Datum trke: "+dan+"."+mesec+"."+godina+" Vreme: "+
            sat+"."+minut+"."+sekunda;
    }
}

package reli.izvestavanje;

public interface IzvestavanjeOReliju {

    public void ispisiNajbrzePosade();

}

package reli;

import java.util.GregorianCalendar;
import java.util.LinkedList;
import reli.izvestavanje.IzvestavanjeOReliju;
import reli.posade.ReliPosada;

public class ReliLista implements IzvestavanjeOReliju{

    private LinkedList <ReliPosada> posade = new LinkedList<ReliPosada>();

    public void unesiUListu(ReliPosada rp){
        if (rp!=null) posade.addFirst(rp);
        else System.out.println("Uneli ste null objekat");
    }

    public void ispisiNajbrzePosade(){
        for (int i=0;i<posade.size();i++){
            ReliPosada rp = posade.get(i);
            GregorianCalendar vreme = rp.getVreme();

            if (vreme.get(GregorianCalendar.HOUR)<1)
                System.out.println(rp);
        }
    }

    public void ispisiPosadeSaIstimPrezimenom(){
        for (int i=0;i<posade.size();i++){
            ReliPosada p = posade.get(i);

            String prezimeV = p.getVozac().substring(p.getVozac().indexOf(' ')+1);
            String prezimeS = p.getSuvozac().substring(p.getSuvozac().indexOf(' ')+1);

            if (prezimeV.equals(prezimeS))
                System.out.println(posade.get(i));
        }
    }
}

```

Zadatak 2

Napraviti javnu apstraktnu klasu **EksplzivnaNaprava** u paketu **naoruzanje** koja ima:

- Privatni atribut naziv.
- Privatni atribut snaga koji predstavlja snagu eksplozivne naprave u megatonima (realan broj).
- Privatni atribut rokTrajanja koji predstavlja datum do kojeg se naprava može upotrebiti.
- Odgovarajuće javne get i set metode za ova dva atributa. Atribut naziv ne sme da bude NULL String, a snaga mora da bude veća od nule. Takođe, rok trajanja mora da se odnosi na neki datum u budućnosti. U slučaju unosa ovih nedozvoljenih vrednosti potrebno je ispisati poruku o grešci.
- Redefinisano metodu toString klase Object koja vraća jedan String u kome se nalaze svi podaci o eksplozivnoj napravi uz odgovarajući tekst.
- Javnu apstraktnu metodu nacinAktiviranja koja nema ulaznih argumenata, a vraća String.

Napraviti javnu klasu **AtomskaBomba** u paketu **naoruzanje.bombe** koja nasleđuje apstraktnu klasu EksplzivnaNaprava i ima:

- Implementiranu apstraktnu metodu nacinAktiviranja koja vraća String sa porukom o tome da se ova bomba aktivira pri kontaktu sa metom.

Napraviti javnu klasu **NuklearniArsenal** u paketu **naoruzanje** koja ima:

- Privatni atribut bombe koji predstavlja niz objekata klase AtomskaBomba.
- Javni konstruktor koji kao ulazni argument prima kapacitet nuklearnog arsenala i postavlja atribut bombe na taj kapacitet ako je uneti broj veći od nule. Ako je uneti broj nula ili manji od nule, kapacitet se postavlja na 5. U svakom slučaju, potrebno je pojedinačno inicijalizovati svaki element niza i to tako da naziv bombe bude “nepoznat”, a snaga 1.
- Javnu metodu unesiBombu koja kao ulazne parametre dobija naziv bombe, njenu snagu i rok trajanja i unosi ih na prvo slobodno mesto u nizu. Ako u nizu nema slobodnih mesta, potrebno je ispisati poruku o tome.
- Javnu metodu ispisiNajjaciArsenal koja na ekranu ispisuje samo podatke o onim bombama čija snaga iznosi više od 5 megatona i kojima još uvek nije istekao rok trajanja.

```
package naoruzanje;

import java.util.GregorianCalendar;

public abstract class EksplzivnaNaprava {

    private String naziv;
    private double snaga;
    private GregorianCalendar rokTrajanja;

    public GregorianCalendar getRokTrajanja() {
        return rokTrajanja;
    }

    public void setRokTrajanja(GregorianCalendar rokTrajanja) {
        GregorianCalendar danasnjiDatum =
            new GregorianCalendar();

        if (rokTrajanja == null || rokTrajanja.before(danasnjiDatum))
            System.out.println("Greska");
        else
            this.rokTrajanja = rokTrajanja;
    }

    public String getNaziv() {
        return naziv;
    }

    public void setNaziv(String naziv) {
        if (naziv == null)
            System.out.println("Greska");
        else
            this.naziv = naziv;
    }
}
```

```

    public double getSnaga() {
        return snaga;
    }

    public void setSnaga(double snaga) {
        if (snaga<=0)
            System.out.println("Greska");
        else
            this.snaga = snaga;
    }

    public String toString(){
        return "Naziv: "+naziv+" Snaga: "+snaga+
            " Rok trajanja: "+rokTrajanja.getTime();
    }

    public abstract String nacinAktiviranja();
}

package naoruzanje.bombe;

import naoruzanje.EksplozivnaNaprava;

public class AtomskaBomba extends EksplozivnaNaprava {

    public String nacinAktiviranja() {
        return "Bomba se aktivira pri kontaktu sa metom";
    }
}

package naoruzanje;

import java.util.GregorianCalendar;
import naoruzanje.bombe.AtomskaBomba;

public class NuklearniArsenal {

    private AtomskaBomba[] bombe;

    public NuklearniArsenal(int brojBombi){
        if (brojBombi>0)
            bombe = new AtomskaBomba[brojBombi];
        else
            bombe = new AtomskaBomba[5];

        for (int i=0; i<bombe.length;i++){
            bombe[i] = new AtomskaBomba();
            bombe[i].setNaziv("nepoznat");
            bombe[i].setSnaga(1);
        }
    }

    public void unesiBombu(String naziv, double snaga, GregorianCalendar rokTrajanja){
        boolean ubacen = false;

        for(int i=0;i<bombe.length;i++){
            if(bombe[i].getNaziv().equals("nepoznat")){
                bombe[i].setNaziv(naziv);
                bombe[i].setSnaga(snaga);
                bombe[i].setRokTrajanja(rokTrajanja);
                ubacen = true;
                break;
            }
        }

        if(!ubacen) System.out.println("U nizu nema mesta");
    }

    public void ispisiNajjaciArsenal(){
        GregorianCalendar danasnjiDatum =
            new GregorianCalendar();

        for (int i=0; i<bombe.length;i++){
            if (bombe[i].getSnaga()>5 &&
                bombe[i].getRokTrajanja().after(danasnjiDatum))
                System.out.println(bombe[i]);
        }
    }
}

```

Zadatak 3

Napraviti javnu klasu **ReliPosada** u paketu **posade** koja ima:

- Privatni atribut **vozac** koji predstavlja ime i prezime vozača dato u formatu "IME PREZIME".
- Privatni atribut **suvozac** koji predstavlja ime i prezime suvozača dato u formatu "IME PREZIME".
- Odgovarajuće javne **get** i **set** metode za ove atribute. Nedozvoljene vrednosti za prvi atribut su **NULL String** i prazan **String** (""), dok drugi atribut ne sme biti **NULL**. U slučaju unosa nedozvoljene vrednosti, potrebno je ispisati poruku o grešci na ekranu.
- Redefinisano metodu **equals** klase **Object** koja kao ulazni argument prima objekat klase **Object**, ali se smatra da će se zaista unositi objekti klase **ReliPosada**. Prvo, potrebno je proveriti da li je zaista unet objekat klase **ReliPosada**, pa ako nije vratiti **false**. Ako je unet objekat klase **ReliPosada**, ova metoda vraća **true** ako je ime i prezime vozača jednako imenu i prezimenu vozača **reli posade** koja je uneta kao ulazni argument. U suprotnom, metoda vraća **false**.

Napraviti javni interfejs **IzvestavanjeOVozacima** u paketu **reli** koji ima:

- Javnu metodu **ispisiSveVozace** koja nema ulaznih argumenata i ne vraća ništa.

Napraviti javnu klasu **ReliLista** u paketu **reli.lista** koja implementira interfejs **IzvestavanjeOVozacima** i ima:

- Privatni atribut **vremeTrke** koji predstavlja datum i vreme kada trka počinje.
- Privatni atribut **posade** koji predstavlja listu objekata klase **ReliPosada**.
- Javni konstruktor koji inicijalizuje atribut **posade** i atribut **vremeTrke** (na trenutni datum i vreme).
- Javnu metodu **unesiUListu** koja kao ulazni argument prima objekat klase **ReliPosada** i unosi ga na kraj liste samo ako uneti objekat nema **null** vrednost. U suprotnom, potrebno je ispisati poruku koja opisuje u čemu je greška.
- Implementiranu javnu metodu **ispisiSveVozace** koja na ekranu ispisuje podatke o svim vozačima i to o svakom vozaču u posebnom redu. Na početku, metoda treba da ispiše podatak o datumu i vremenu trke uz odgovarajuću poruku.
- Javnu metodu koja na osnovu postojeće liste **posada** pravi i vraća novi objekat klase **ReliLista** u kojem se kao **reli posade** nalaze samo one **posade** iz postojeće liste u kojima nema suvozača (ime suvozača je **null** ili prazan **String**).

```
package posade;

public class ReliPosada {

    private String vozac;
    private String suvozac;

    public String getVozac() {
        return vozac;
    }

    public void setVozac(String vozac) {
        if (vozac == null || vozac.equals(""))
            System.out.println("Greska");
        else
            this.vozac = vozac;
    }

    public String getSuvozac() {
        return suvozac;
    }

    public void setSuvozac(String suvozac) {
        if (suvozac == null)
            System.out.println("Greska");
        else
            this.suvozac = suvozac;
    }
}
```

```

        public boolean equals (Object o){
            if (!(o instanceof ReliPosada))
                return false;

            ReliPosada rp = (ReliPosada)(o);

            if (vozac.equals(rp.getVozac()))
                return true;
            else
                return false;
        }
    }

    package reli;

    public interface IzvestavanjeOVozacima {

        public void ispisiSveVozace();
    }

    package reli.lista;

    import java.util.GregorianCalendar;
    import java.util.LinkedList;

    import posade.ReliPosada;
    import reli.IzvestavanjeOVozacima;

    public class ReliLista implements IzvestavanjeOVozacima{

        private GregorianCalendar vremeTrke;
        private LinkedList <ReliPosada> posade;

        public ReliLista(){
            posade = new LinkedList<ReliPosada>();
            vremeTrke = new GregorianCalendar();
        }

        public void unesiUListu(ReliPosada rp){
            if (rp!=null)
                posade.addLast(rp);
            else
                System.out.println("Uneli ste null objekat");
        }

        public void ispisiSveVozace(){
            System.out.println("Datum i vreme trke: "+vremeTrke.getTime());

            for(int i=0;i<posade.size();i++)
                System.out.println(posade.get(i).getVozac());
        }

        public ReliLista vratiListuSoloEkipa(){
            ReliLista rl = new ReliLista();

            for(int i=0;i<posade.size();i++)
                if(posade.get(i).getSuvozac()==null ||
                    posade.get(i).getSuvozac().equals(""))
                    rl.unesiUListu(posade.get(i));

            return rl;
        }
    }

```

Zadatak 4

Napraviti javnu apstraktnu klasu **EksplzivnaNaprava** u paketu **eksplozivne_naprave** koja ima:

- Privatni atribut naziv.
- Privatni atribut snaga koji predstavlja snagu eksplozivne naprave u megatonima (realan broj).
- Odgovarajuće javne get i set metode za ova dva atributa. Atribut naziv ne sme da bude NULL String niti prazan String (""), a snaga mora da bude veća od nule. U slučaju unosa ovih nedozvoljenih vrednosti potrebno je ispisati poruku o grešci.
- Redefinisanu metodu equals klase Object koja kao ulazni argument prima objekat klase Object, ali se smatra da će se zaista unositi objekti klase EksplzivnaNaprava. Ova metoda vraća true ako je naziv naprave jednak nazivu eksplozivne naprave koja je uneta kao ulazni argument. U suprotnom, metoda vraća false.
- Javnu apstraktnu metodu dozvoljenoAktiviranje koja nema ulaznih argumenata, a vraća boolean vrednost.

Napraviti javnu klasu **InterkontinentalnaRaketa** u paketu **naoruzanje** koja nasleđuje apstraktnu klasu EksplzivnaNaprava i ima:

- Privatni atribut vremeLansiranja koji predstavlja datum i vreme kada je potrebno lansirati projektil.
- Javnu set metodu za ovaj atribut (**bez get metode**). Nedozvoljene vrednosti za ovaj atribut su NULL vrednosti. U slučaju unosa ovih vrednosti ispisati poruku o grešci.
- Implementiranu apstraktnu metodu dozvoljenoAktiviranje koja vraća vrednost TRUE ako je trenutni datum i vreme posle vremena lansiranja, a inače FALSE.

Napraviti javnu klasu **InterkontinentalniArsenal** u paketu **naoruzanje** koja ima:

- Privatni atribut rakete koji predstavlja niz objekata klase InterkontinentalnaRaketa.
- Javni konstruktor koji kao ulazni argument prima kapacitet nuklearnog arsenala i postavlja atribut rakete na taj kapacitet ako je uneti broj veći od nule. Ako je uneti broj nula ili manji od nule, kapacitet se postavlja na 5.
- Javnu metodu unesiRaketu koja kao ulazne parametre dobija objekat klase InterkontinentalnaRaketa i unosi ga na prvo slobodno mesto u nizu. Smatra se da je mesto slobodno ako element na tom mestu ima vrednost null. Ako u nizu nema slobodnih mesta, potrebno je ispisati poruku o tome.
- Javnu metodu ispisiArsenalSrednjeJacine koja na ekranu ispisuje samo nazive onih raketa čija snaga iznosi između 3 i 5 megatona (uključujući i te vrednosti) i koje mogu da se aktiviraju.

```
package eksplozivne_naprave;

public abstract class EksplzivnaNaprava {

    private String naziv;
    private double snaga;

    public String getNaziv() {
        return naziv;
    }

    public void setNaziv(String naziv) {
        if (naziv==null || naziv.equals(""))
            System.out.println("Greska");
        else
            this.naziv = naziv;
    }

    public double getSnaga() {
        return snaga;
    }

    public void setSnaga(double snaga) {
        if (snaga<=0)
            System.out.println("Greska");
        else
            this.snaga = snaga;
    }
}
```

```

    }

    public boolean equals (Object o){
        EksplozivnaNaprava en = (EksplozivnaNaprava)(o);

        if (naziv.equals(en.getNaziv()))
            return true;
        else
            return false;
    }

    public abstract boolean dozvoljenoAktiviranje();
}

package naoruzanje;

import java.util.GregorianCalendar;
import eksplozivne_naprave.EksplozivnaNaprava;

public class InterkontinentalnaRaketa extends EksplozivnaNaprava {

    private GregorianCalendar vremeLansiranja;

    public void setVremeLansiranja(GregorianCalendar vremeLansiranja) {
        if (vremeLansiranja == null)
            System.out.println("Greska");
        else
            this.vremeLansiranja = vremeLansiranja;
    }

    public boolean dozvoljenoAktiviranje() {
        if (vremeLansiranja.after(new GregorianCalendar()))
            return true;
        else
            return false;
    }
}

package naoruzanje;

public class InterkontinentalniArsenal {

    private InterkontinentalnaRaketa[] rakete;

    public InterkontinentalniArsenal(int brojRaketa){
        if (brojRaketa>0)
            rakete = new InterkontinentalnaRaketa[brojRaketa];
        else
            rakete = new InterkontinentalnaRaketa[5];
    }

    public void unesiRaketu (InterkontinentalnaRaketa ir){
        boolean uneta = false;

        for(int i=0;i<rakete.length;i++){
            if (rakete[i]==null){
                rakete[i]=ir;
                uneta = true;
                break;
            }
        }

        if(!uneta) System.out.println("Niz je pun");
    }

    public void ispisiArsenalSrednjeJacine(){
        for(int i=0;i<rakete.length;i++){
            if (rakete[i]!=null &&
                rakete[i].getSnaga()>=3 &&
                rakete[i].getSnaga()<=5 &&
                rakete[i].dozvoljenoAktiviranje())
                System.out.println(rakete[i].getNaziv());
        }
    }
}

```