PROJECT ΟΝΤΟΚΕΝΤΡΙΚΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ 2020

AM:AM1084576

ΟΝΟΜΑΤΕΠΩΝΥΜΟ:Μαριάνθη Θώδη

Email:up1084576@upatras.gr


```
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;


public class Main extends Menu {
    public static void main(String[] args) throws ExceptionO, ExceptionR {
        ArrayList<Material> MaterialList = new ArrayList<>();
        ArrayList<Service> ServiceList = new ArrayList<>();
        ArrayList<RequestDonation> Alltogether = new ArrayList<>();
        Admin Marianthi = new Admin("Marianthi", "6946930521", true);
        Organization Donations = new Organization("Donations", Marianthi);
        Donations.setAdmin(Marianthi);
        Donations.listofphones("6946930521");
        Beneficiary ben1 = new Beneficiary("Olga", "6948063296", 4);
        Donations.insertBeneficiary(ben1);
        Donations.listofphones("6948063296");
        Beneficiary ben2 = new Beneficiary("Marina", "6948063297", 5);
        Donations.listofphones("6948063297");
        Donations.insertBeneficiary(ben2);


        Material milk = new Material("milk ", "drink", 1);
        RequestDonation milk1 = new RequestDonation(milk, 10);
        MaterialList.add(milk);
        Alltogether.add(milk1);
        milk1.setName("milk");
        milk1.setId(1);
        ben1.add(milk1);
        milk1.setDescription("  Milk is a nutrient mixture of various organic
substances and consists of water, fat, proteins, carbohydrates, enzymes, salts and
vitamins.");
        Material rice = new Material("rice", "eat ", 2);
        RequestDonation rice1 = new RequestDonation(rice, 10);
        MaterialList.add(rice);
        Alltogether.add(rice1);
        rice1.setName("rice");
        rice1.setId(2);
        ben2.add(rice1);
        rice1.setDescription(" Rice is one of the staple foods of mankind");
        Material sugar = new Material("sugar", "eat ", 3);
        RequestDonation sugar1 = new RequestDonation(milk, 10);
        MaterialList.add(sugar);
        Alltogether.add(sugar1);
        sugar1.setName("sugar");
        sugar1.setDescription(" To make delicious sweets!");
```

```
sugar1.setId(3);
Service medicalSupport = new Service("MedicalSupport", "service ", 4);
RequestDonation medicalSupport1 = new RequestDonation(medicalSupport, 10);
ServiceList.add(medicalSupport);
Alltogether.add(medicalSupport1);
medicalSupport.setDescription(" Medical support provides either cash medical
support or health insurance.");
medicalSupport1.setName("medical Support");
medicalSupport1.setId(4);
Service nurseySupport = new Service(" NurserySupport", "service ", 5);
RequestDonation nurseySupport1 = new RequestDonation(nurseySupport, 10);
ServiceList.add(nurseySupport);
Alltogether.add(nurseySupport1);
nurseySupport1.setId(5);
nurseySupport1.setName(" Nursey Support");
nurseySupport1.setDescription(" Provides specialized hospitals");
Service babySitting = new Service("BabySitting", "service ", 6);
RequestDonation babysitting1 = new RequestDonation(babySitting, 10);
ServiceList.add(babySitting);
Alltogether.add(babysitting1);
babysitting1.setName("baby sitting");
babysitting1.setDescription(" Provides nannies who love little human
creatures!");
babysitting1.setId(6);
Donator nikol=new Donator("Nikol Papagiannakopoulou" , "6981753467");
Donations.insertDonator(nikol);
nikol.add1(milk1);


String name1;
int atoma;
Scanner sc = new Scanner(System.in);
System.out.println("Give your phone:");
String phone = sc.next(); //διαβαζει το τηλεφωνο του χρηστη

if (Marianthi.getPhone().equals(phone)) {
System.out.println("You are the admin!");
itsAdmin(Marianthi);
ItsMenuForAdmin();
System.out.println("Select  by writing the number of your choice:");
Scanner cc = new Scanner(System.in);
int c = cc.nextInt();
if (c == 1) {
babysitting1.print(Alltogether);
System.out.println("Choose a product that you want to learn more
about: ");
Scanner cc1 = new Scanner(System.in);
int b = cc1.nextInt();
System.out.println(Alltogether.get(b).getEntityInfo());

}
if (c == 2) {
ItsMenuforAdminMO();
System.out.println("Choose  that you want to learn more about: ");
Scanner cc1 = new Scanner(System.in);
int b = cc1.nextInt();
if (b == 1) {
```

```
                    listBeneficiary();
                System.out.println("Select a beneficiary: ");
                    Scanner cc2 = new Scanner(System.in);
                            String d = cc2.next();
                  Beneficiary a = searchforBeneficiary(d);
                            a.printReceivedlist();
                 System.out.println("1. Clear  its benefits.");
                System.out.println("2. Remove the beneficiary.");
                        System.out.println("Choose:");
                    Scanner cc3 = new Scanner(System.in);
                            int p = cc2.nextInt();
                                if (p == 1) {
                                    a.reset();
                                    }
                                if (p == 2) {
                            Donations.removeBeneficiary(a);
                                    }

                                }else if(b==2){
                                  listDonator();
                        System.out.println("Select a Donator: ");
                          Scanner cc2 = new Scanner(System.in);
                                  String d = cc2.next();
              System.out.println("1.The current status of the benefits it
                intends to  offers the Donator.");
                     System.out.println(" 2.Delete the donator");
                       System.out.println("Choose 1 or 2: " );
                        Scanner cc4 = new Scanner(System.in);
                                int z = cc4.nextInt();
                           Donator a = searchforDonator(d);
                                    if(z==1){
                                a.printOfferlist(a);}
                                if (z == 2) {
                            Donations.removeDonator(a);
                                    }

                                }


        } else if (!(Donations.searchForPhone(phone))) {
  System.out.println("Do you want to join? If yes write 1 else 2: ");
                            int a = sc.nextInt();
                                if (a == 1) {
          System.out.println("Do you want to be a Beneficiary or a
Donator?If you want the first write 1 else write 2:");
                            int b = sc.nextInt();
                                if (b == 1) {

            System.out.println("give me the number of people in your
                            family:");
          atoma = sc.nextInt();//αριθμος ατομων που εχει η οικογενεια
                                του
                      System.out.println("Give me your name:");
                                    name1 = sc.next();
                Beneficiary ben3 = new Beneficiary(name1, phone, atoma);
                                ben3.setnoPerson(atoma);
                            Donations.insertBeneficiary(ben3);
```

```
                itsBeneficiary(ben3);

            } else if (b == 2) {
        System.out.println("Give me your name:");
                name1 = sc.next();
        Donator ben3 = new Donator(name1, phone);
            Donations.insertDonator(ben3);
                itsDonator(ben3);

                    }
                } else if (a == 2) {
                    System.exit(0);
                        }
                    }

                }
            }
        }


public class Admin extends User{
        boolean isAdmin=true;

            Admin(){

                }


    Admin( String nameuser , String phone, boolean isAdmin ){
                super(nameuser ,phone);
                this.isAdmin=isAdmin;
                    }

            public boolean isAdmin() {
                return isAdmin;
                    }
        public void setAdmin(boolean admin) {
                isAdmin = admin;
                    }
                }


                ;

            import java.util.ArrayList;

        public  class Beneficiary extends  User{
                private   int noPerson ;
  private static  ArrayList<RequestDonation> receivedList = new ArrayList<>();//υ
    αναπαριστά τη λίστα των ειδών και των ποσοτήτων που έχει ήδη λάβει.
   private static  ArrayList <Request> requestsList = new ArrayList<>();// που
αναπαριστά την τρέχουσα λίστα των ειδών και των ποσοτήτων που ζητά να του
                δοθούν.
```

```
Beneficiary(String nameuser ,String phone , int noPerson){
                    super(nameuser ,phone);
                    setnoPerson(noPerson);
                            }



                    public  int getnoPerson() {
                          return this.noPerson;
                            }


                public void setnoPerson(int noPerson) {
                            this.noPerson = noPerson;
                            }

            public static int Checkingforquantity(Beneficiary a) {
                if (a.getnoPerson() >= 2 && a.getnoPerson() <= 4) {
                        System.out.println("You are at level2");
                                    return 2;
                         } else if (a.getnoPerson() == 1) {
                        System.out.println("You are at level1");
                                    return 1;
                         } else if (a.getnoPerson() == 5) {
                        System.out.println("You are at level3");
                                    return 3;
                                      }

                                   return 0;
                                    }

             public static ArrayList<Request> getRequestsList() {
                                    return requestsList;
                                    }

        public static void setRequestsList(ArrayList<Request> requestsList) {
                        Beneficiary.requestsList = requestsList;
                                    }

            public static ArrayList<RequestDonation> getReceivedList() {
                                    return receivedList;
                                    }

        public  void setReceivedList(ArrayList<RequestDonation> receivedList) {
                        Beneficiary.receivedList = receivedList;
                                    }


                        public void add(RequestDonation a ){
                               getReceivedList().add( a);
                                    }

                        public void remove(RequestDonation a){
                               getReceivedList().remove(a);
```

```
    }

    public void reset(){
    getReceivedList().clear();
    }



public void modify(RequestDonation r , double quantity){
        int a=getReceivedList().indexOf(r);
            quantity=r.getQuantity();
    getReceivedList().set(a , (RequestDonationList) r);

    }

    public static void  printReceivedlist(){
    for(int i=0; i<getReceivedList().size(); i++){
       System.out.println( (i+1)+"." +" Entity: " +
    getReceivedList().get(i).getName());
            System.out.println("Quantity: " +
    getReceivedList().get(i).getQuantity());}

    }

    }



    import java.util.ArrayList;

    class Donator extends User{
private static ArrayList <Offers> offerlist=new ArrayList<>(); // το οποίο
    αναπαριστά τη λίστα ειδών που επιθυμεί να προσφέρει

    Donator(String nameuser ,String phone ){
        super(nameuser, phone);
    }

    public ArrayList<Offers> getOfferlist() {
        return offerlist;
    }

public void setOfferlist(ArrayList<Offers> offerlist) {
        Donator.offerlist = offerlist;
    }

    public  void add1(Offers a ){
        getOfferlist().add(a);

    }

    public  void  printOfferlist(){
    for(int i=0; i<getOfferlist().size(); i++){
```

```
                System.out.println( (i+1)+"." +" Entity: " +
               getOfferlist().get(i).getName());
        System.out.println("Quantity: " + getOfferlist().get(i).getQuantity());}
                                    }


            public void modify(Offers r , double quantity){
                    int a=getOfferlist().indexOf(r);
                        quantity=r.getQuantity();
                   getOfferlist().set(a , (Offers) r);

                                    }
                                }


          // αντιπροσωπεύει ένα είδος δωρεάς
                abstract class Entity {
                    private String name;
                 private String description;
                       private int id;

                          Entity(){

                                    }

                        Entity(Entity entity) {

                                    }


            Entity(String name, String description, int id) {
                              this.name = name;
                        this.description = description;
                                 this.id = id;
                                    }


                        public String getEntityInfo() {
                                return toString();
                                    }

public  String toString( ){return "The product is: " + getName() + " ,the id is:
       " +getId() + " and the description is: " + getDescription();}

                         public String getName() {
                                 return name;
                                    }

                            public int getId() {
                                   return id;
                                    }

                        public void setId(int id) {
```

```
        this.id = id;
    }

    public String getDescription() {
        return description;
    }

    public void setDescription(String description) {
        this.description = description;
    }

    public void setName(String name) {
        this.name = name;
    }

    protected abstract int getLevel();
}


public class Material extends  Entity {
    static protected int level;
    static protected int human;


Material(String name, String description, int id) {
    super(name, description, id);
    }

    @Override
    public String toString() {
        return null;
    }


    Material() {
    }


    public int getLevel() {
        return level;
    }

    public void setLevel(int level) {
        this.level = level;
    }


public class Menu extends Organization {

    public static void itsDonator(Donator a) {
System.out.println("Hello  member , we are the Donations!");
        System.out.println(a.getUserInfo());

    }
```

```
    public static void itsBeneficiary(Beneficiary a) {
        System.out.println("Hello  member, we are the Donations!");
                System.out.println(a.getUserInfo());

                }

    public static void itsAdmin(Admin a) {
        System.out.println("Hello  member, we are the Donations!");
                System.out.println(a.getUserInfo());

                }

    public static void ItsMenuForAdmin() {
                System.out.println("MENU:");
        System.out.println("1.View" + '\n' + "2.Monitor Organization" + '\n' +
                "3.Back" + '\n' + "4.Logout" + '\n' + "5.Exit");

                }


    public static void ItsMenuforAdminMO() {
                System.out.println("Monitor Organization!");
        System.out.println("1.List Beneficiaries " + '\n' + "2.List Donators " + '\n'
                + "3.Reset Beneficiaries Lists ");

                }


    public static Beneficiary searchforBeneficiary(String s)
                {
        for (int i = 0; i < getBeneficiaryList().size(); i++)
        if (s.equals(getBeneficiaryList().get(i).getNameuser())){
                    return getBeneficiaryList().get(i);
                            }
                    return null;
                }

    public static Donator searchforDonator(String s)
                {
      for (int i = 0; i < getDonatorList().size(); i++)
      if (s.equals(getDonatorList().get(i).getNameuser())){
                    return getDonatorList().get(i);
                            }
                    return null;


                }
            }


import java.util.ArrayList;

public class Offers extends RequestDonationList { // Αναπαριστά το σύνολο των ειδών
που προτίθεται να συνεισφέρει ο Donator.
```

```
    Offers(Entity entity , int quantity , int quantity1){
        super(entity , quantity );

    }


public void commit( ArrayList<RequestDonation> rdEntities){
        rdEntities=super.getRdEntities();

    }

    @Override
public void modify(RequestDonation r, int quantity) {
        super.modify(r,  quantity);
    }

    @Override
public void add(RequestDonation r) throws ExceptionR {
        super.add(r);
    }

    @Override
public String monitor() {
    return super.monitor();
    }
  }

import java.util.*;

public class Organization {
// Αντιστοιχεί στον οργανισμό που υποστηρίζει το σύστημα donation.
    private String name;
    private Admin admin;
private static ArrayList <Entity> entityList=new ArrayList<>();//με τα είδη  που μπορούν να διανεμηθούν
    private static ArrayList <RequestDonationList> currentDonations=new ArrayList<>();//διαθέσιμες προσφορές και τις ποσότητές τους
    private static ArrayList<String> phones=new ArrayList<>();
private static ArrayList<Beneficiary> beneficiaryList= new ArrayList<>(); //μια λίστα με τους επωφελούμενους beneficiaryList
private static ArrayList<Donator> donatorList= new ArrayList<>(); //μια λίστα με τους δωρητές


public Admin getAdmin() {
        return admin;
    }

public void setAdmin(Admin admin) {
        this.admin = admin;
    }

public String getName() {
        return name;
```

```
}


Organization(){

}

Organization(String name , Admin admin)
{
this.name=name;
this.admin=admin;
}


public void addEntity(Entity k)  {
entityList.add(k);
}


public void removeEntity(Entity ob) {
entityList.remove(ob);
}

public void insertDonator(Donator d) throws ExceptionO
{
donatorList.add(d);

}

public void removeDonator(Donator a){
donatorList.remove(a);
}

public  void insertBeneficiary(Beneficiary a){
beneficiaryList.add(a);

}

public void removeBeneficiary(Beneficiary a){
beneficiaryList.remove(a);
}


public static ArrayList<Entity> getEntityList() {
return entityList;
}

public static void setEntityList(ArrayList<Entity> entityList) {
Organization.entityList = entityList;
}

public  static void listEntities(){
for(int i=0; i<entityList.size(); i++)
System.out.println ((i+1)+"."+getEntityList().get(i).toString());
```

```
}

    public static void listDonator(){
        System.out.println("List of Donators:");
        for(int i=0; i<donatorList.size(); i++ ){
    System.out.println( (i+1)+ "." +donatorList.get(i).getNameuser());
                }

    }

    public static void listBeneficiary(){
        System.out.println("List of Beneficiaries:");
        for(int i=0; i<beneficiaryList.size(); i++ ){
    System.out.println((i+1)+ "." +beneficiaryList.get(i).getNameuser());
                }
    }

    public static boolean searchForPhone(String phone ) {
            for(int i=0;i<getPhones().size(); i++){
              if(getPhones().get(i).equals(phone))
                        return true;}

                    return false;
    }

    public void  listofphones(String phone){
                    phones.add(phone);
    }

    public static ArrayList<String> getPhones() {
                    return phones;
    }



    public static ArrayList<Beneficiary> getBeneficiaryList()  {
                    return beneficiaryList;
    }

    public static ArrayList<Donator> getDonatorList() {
                    return donatorList;
    }

public static ArrayList<RequestDonationList> getCurrentDonations() {
                    return currentDonations;
    }
}
import java.util.Scanner;
public class Request extends RequestDonationList {
    Request(Entity entity, int quantity) {
            super(entity, quantity);
    } //Αναπαριστά το σύνολο των ειδών

    public Request(Material entity) {
    }
```

```
// (Material ή Services) που ζητά ο Beneficiary //

                @Override
    public void modify(RequestDonation r, int quantity) {
                super.modify(r, quantity);
                }

                @Override
    public void add(RequestDonation r) throws ExceptionR {
                super.add(r);
                }


                }

            import java.util.ArrayList;
            import java.util.Comparator;
public class RequestDonation  extends Entity implements Comparator<Entity>  {

                private Entity entity;
                private int quantity;

            RequestDonation(Entity entity, int quantity){
                super(entity);
            this.quantity=quantity;
                }


                RequestDonation(){

                }


        public int compare (Entity ob1, Entity ob2){
                int idCompare1 = ob1.getId();
                int idCompare2 = ob2.getId();
                if (idCompare1 == idCompare2)
            System.out.println("Its the same item");
                        else
            System.out.println("Its different item");
                return 0;

                }


            public Entity getEntity() {
                return entity;
                }

        public void setEntity(Entity entity) {
                this.entity = entity;
                }

            public int getQuantity () {
                return quantity;
```

```
    }

    public void setQuantity ( int quantity){
        this.quantity = quantity;
    }

    @Override
    public String getName() {
        return super.getName();
    }

    @Override
    protected int getLevel() {
        return getLevel();
    }


    @Override
    public String toString() {
return ("The product is: " + getName() + " ,the id is: " +getId() + " and the
                description is: " + getDescription());
    }

    public  String print(ArrayList<RequestDonationList> str){
        RequestDonation[] arr=new RequestDonation[str.size()];
System.out.println("list of donor names and existing quantities");
        str.toArray(arr);
        for (int i=0; i< arr.length; i++){
        System.out.println("Name of entity : " +arr[i].getName());
      System.out.println("Quantity of entity :" + arr[i].getQuantity());

        }
    return  null;

    }

}

import java.util.ArrayList;
 import java.util.Scanner;

public class RequestDonationList extends  RequestDonation{
//Αναπαριστά μια συλλογή από αντικείμενα RequestDonation και δίνει μεθόδους
                για την ενημέρωση της λίστας αυτής
    private ArrayList <RequestDonation> rdEntities;

    RequestDonationList(){

    }

    RequestDonationList(Entity entity, int quantity) {
        super(entity, quantity);
    }
```

```
public ArrayList<RequestDonation> getRdEntities() {
return rdEntities;
}

public void setRdEntities(ArrayList<RequestDonation> rdEntities) {
this.rdEntities = rdEntities;
}


public void add(RequestDonation r) throws ExceptionR {
if (rdEntities.contains(r)) {
int thesi = rdEntities.indexOf(r);
System.out.println("Give the quantity that you will be donate:");
Scanner sc = new Scanner(System.in);
int quantity = sc.nextInt();
modify(r, quantity);
} else {
System.out.println("Do you want to add it?Yes or no");
Scanner sc1 = new Scanner(System.in);
String answer = sc1.next();
if (answer.equals("yes")) {
try {
rdEntities.add(r);
} catch (Exception e) {
throw new ExceptionR(e.toString());
}
}else{}
}
}


public void remove(RequestDonation r){
rdEntities.remove(r);
}

public void modify(RequestDonation r , int quantity){
int a=getRdEntities().indexOf(r);
r.setQuantity(r.getQuantity()+quantity);
getRdEntities().set(a , r);


}

public String monitor(){
RequestDonation[] arr=new RequestDonation[rdEntities.size()];
System.out.println("list of donor names and existing quantities");
rdEntities.toArray(arr);
for (int i=0; i< arr.length; i++){
System.out.println("Name of entity : " +arr[i].getName());
System.out.println("Quantity of entity :" + arr[i].getQuantity());

}
return null;
```

```
}


    public  boolean itsMaterialorService(RequestDonation r){
            if(r.getLevel() == Integer.parseInt(null))
                        return true;
                    else return false;
                        }

public void validRequestDonation(Beneficiary a , Request b) {
                    if(itsMaterialorService(b))
                System.out.println("Its Service item");
                            else {
                    System.out.println("Its Material Item");
                        int m=a.Checkingforquantity(a);
                      Scanner sc=new Scanner(System.in);
                                if(m==1)
                                    {
        System.out.println("Give the quantity that you want:");
                            int quantity=sc.nextInt();
                           if(quantity>getQuantity()){
           System.out.println("Your request has been rejected");
                                    }
                                else
                            quantity=getQuantity();
                                    }
                                if(m==2)
                                    {
        System.out.println("Give the quantity that you want:");
                            int quantity=sc.nextInt();
                            if(quantity>getQuantity())
           System.out.println("Your request has been rejected");
                                else
                            quantity=getQuantity();
                                    }
                                if(m==3)
                                    {
                System.out.println("You are at a level3");
        System.out.println("Give the quantity that you want:");
                            int quantity=sc.nextInt();
                            if(quantity>getQuantity())
           System.out.println("Your request has been rejected");
                                else
                            quantity=getQuantity();
                                    }
                                 }


                                }


                                }
                    public class Service extends Entity {
            Service(String name, String description, int id) {
```

```
        super(name, description, id);
                }
           String getDetails(){
               return toString();
                }


                @Override
        public String toString() {
         return "the item is Service";
                }

                @Override
        protected int getLevel() {
                  return 0;
                }
               }
         abstract class User {
          private String nameuser;
            private String phone;


                    User(){

                    }
          User(String nameuser ,String phone){
                 this.nameuser=nameuser;
                    this.phone=phone;
                    }
              public String  getPhone() {
                    return phone;
                    }

           public void setPhone(String phone) {
                   this.phone = phone;
                    }

               public String getNameuser() {
                    return nameuser;
                    }

        public void setNameuser(String nameuser) {
                   this.nameuser = nameuser;
                    }

                public String getUserInfo(){
return ("Name: "+ getNameuser() + "Phone :" + getPhone() );
                    }
                   }
```