ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
UNIVERSITY OF PIRAEUS

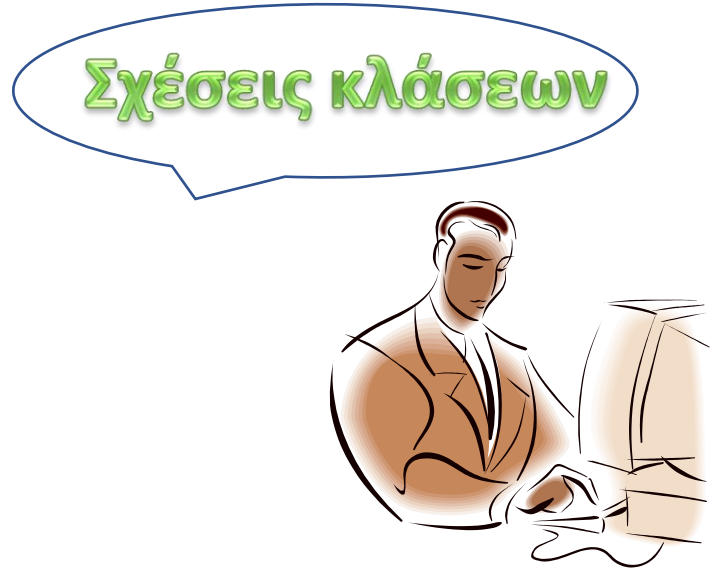
Αντικειμενοστρεφής Προγραμματισμός
Ασκήσεις
11/6/2024

Εαρινό Εξάμηνο 2024

Κούτσικας Χρήστος

Άσκηση 1

α) Η κλάση BankAccount μοντελοποιεί ένα τραπεζικό λογαριασμό. Διαθέτει τον αριθμό του τραπεζικού λογαριασμού και μια λίστα από κινήσεις (κλάση Trn). Οι κινήσεις διακρίνονται σε χρεώσεις (αναλήψεις) και πιστώσεις (καταθέσεις). Υλοποιείστε τις κλάσεις.



Άσκηση 1

```
class BankAccount {
   String bankNumber;
   ArrayList<Trn> trns;

   BankAccount (String b){
      bankNumber=b;
      trns= new ArrayList<Trn>();
   }
}

abstract class Trn {
   BankAccount bankAccount;
   //LocalDate date;
   double val;

   Trn (){}

   Trn (BankAccount b, double v) {
      bankAccount=b;
      val=v;
   }
}

class Debit extends Trn {
   Debit (BankAccount b, double v) {
      super(b,v);
      b.trns.add(this);
   }
}

class Credit extends Trn {
   Credit (BankAccount b, double v) {
      super(b,v);
      b.trns.add(this);
   }
}
```

β) Υλοποιείστε μέθοδο getBal για τον υπολογισμό του υπολοίπου.

Άσκηση 1

```
class BankAccount {

   //double bal;

   double getBal(){
      double b=0.0;
      for (Trn t: trns){
         if (t instanceof Credit)
            b+=t.val;
         else
            b-=t.val;
         }
         //b+=t.getVal();
      return b;
   }
}
```

Ιδιότητα ή μέθοδος;

Άσκηση 1 – Εναλλακτική λύση

```
class BankAccount {

   //double bal;

   double getBal(){
      double b=0.0;
      for (Trn t: trns){
         b+=t.getVal();
      }
      return b;
   }
}

abstract class Trn {
   public abstract double getVal();
}

class Debit extends Trn {
   public double getVal(){return -val;}
}

class Credit extends Trn {
   public double getVal(){return val;}
}
```

Άσκηση 1 – Λύση με interface

```
interface ITrn {
    public double getVal();
}

class BankAccount {

   ArrayList<ITrn> trns= new ArrayList<ITrn>();

   double getBal(){
      double b=0.0;
      for ((ITrn  t: trns){
         b+=t.getVal();
      }
      return b;
   }
}

abstract class Trn implements ITrn {
   //public abstract double getVal();
}

class Debit extends Trn {
   public double getVal(){return -val;}
}

class Credit extends Trn {
   public double getVal(){return val;}
}
```

Άσκηση 1

```
interface ITrn {
    public double getVal();
}

class BankAccount {

   ArrayList<ITrn> trns= new ArrayList<ITrn>();

   double getBal(){
      double b=0.0;
      for ((ITrn  t: trns){
         b+=t.getVal();
      }
      return b;
   }
}

abstract class Trn implements ITrn {
   //public abstract double getVal();
}

class Debit extends Trn {
   public double getVal(){return -val;}
}

class Credit extends Trn {
   public double getVal(){return val;}
}
```

β) Τροποποιείστε τις κλάσεις των κινήσεων έτσι ώστε να προκαλείται exception στην περίπτωση ανάληψης ποσού μεγαλύτερου από το υπόλοιπο

Άσκηση 1

```
abstract class Trn implements ITrn {
   Trn (BankAccount b, double v) {
      bankAccount=b;
      val=v;
   }
}

class Debit extends Trn {
   Debit (BankAccount b, double v) throws Exception{
      super(b,v);
      if (v>b.getBal())
         throw new Exception("Error");
      b.trns.add(this);
   }
}

class Credit extends Trn {
   Credit (BankAccount b, double v) throws Exception {
      super(b,v);
      b.trns.add(this);
   }
}
```

Άσκηση 1 – Εναλλακτική λύση

```
abstract class Trn implements ITrn {
   Trn (BankAccount b, double v) throws Exception{
      bankAccount=b;
      val=v;
      if ((this instanceof Debit) && (v>b.getBal()))
         throw new Exception("Error");
      }
}

class Debit extends Trn {
   Debit (BankAccount b, double v) throws Exception{
      super(b,v);
      b.trns.add(this);
   }
}

class Credit extends Trn {
   Credit (BankAccount b, double v) throws Exception {
      super(b,v);
      b.trns.add(this);
   }
}
```

Άσκηση 1 – Η main

```
public static void main(String[] args) {
   BankAccount b = new BankAccount("LOG");
   Trn t2=null;
   try {
      Trn t1=new Credit(b,100);
      t2=new Debit(b,140);
   }
   catch (Exception e) {
      System.out.println(e.getMessage());
   };

   System.out.println(b.getBal());
}
```