Exercises 17

references

Exercise 17.1 – Drawing data structures

For this exercise, you will draw data structures with small patterns .

Question 17.1.1 - Arrays

Draw each array of the following program as they are at the end of the execution of this program.

```
class Exo_16_1_1{
    public static void main(String[] args){
        int[] t1, t2, t3, t4, t5;
        t2 = new int[4];
        t3 = new int[4];
        t4 = new int[4];
        for (int i=0; i<t3.length; i++){
            t3[i] = 10+i;
            t4[i] = 10+i;
        }
        t5 = t4;
    }
}
```

Question 17.1.2 - Objects

Same question for the program variables as follows:

(var1, var2, var3, cou1, cou2, cou3, cou4, cou5, cou6).

```
class Exo_16_1_2{
    public static void main(String[] args){
        C1 var1, var2, var3;
        C2 cou1, cou2, cou3, cou4, cou5, cou6;
        var1 = new C1(0);
        var2 = new C1(1);
        cou2 = new C2();
        cou3 = new C2();
        cou4 = new C2();
        cou6 = new C2();
        cou3.premier = var1;
        cou3.second = var2;
        cou4.premier = var1;
        cou4.second = var1;
        cou5 = cou3;
        cou6.premier = var1;
        cou6.second = new C1(2);
    }
}
class C1{
    int x;
    C1(int i){
        x=i;
    }
}
class C2{
    C1 premier, second;
}
```

Question 17.1.3 – Arrays of objects

Same question for the program variables as follows:

```
class Exo_16_1_3{
    public static void main(String[] args){
        C1[] t1, t2, t3;
        C1 var = new C1(0);
        t1=new C1[3];
        t2=new C1[3];
        t3=new C1[3];
        for (int i=0; i<t1.length; i++){
            t1[i] = new C1(0);
            t2[i] = var;
        }
        t1[1].x = 1;
        t2[1].x = 1;
        t3 = t2;
    }
}
```

Question 17.1.4 – Objects containing arrays

Same question for the program variables as follows:

```
class Exo_16_1_4{
    public static void main(String[] args){
        C3 pl1, pl2, pl3;
        C1 cl1;
        cl1 = new C1(0);
        pl1 = new C3();
        pl2 = new C3();
        pl3 = new C3();
        pl2.tab = new C1[3];
        pl3.tab = new C1[3];
        pl3.tab[0] = cl1;
        pl3.tab[1] = cl1;
    }
}
class C3{
    C1[] tab;
}
```

Exercise 17.2 - Bank accounts

Represent bank accounts and account holders. A single account can have multiple owners (this is the case, for example, joint accounts for a couple). In this case, the same object will be shared among the owners. A holder can have multiple accounts.
We offer the following skeleton class to implement accounts and holders.

```
class TableauCompte{
    Compte[] tab;
    int longueur;
    TableauCompte(int n){
        tab = new Compte[n];
    }
    void ajouter(Compte c) throws NonInitialise{
        if (c == null){
            throw new NonInitialise();
        }
        if (longueur < tab.length){
            tab[longueur]=c;
            longueur++;
        }
    }
}
class Banque{
    String nom;
    TableauCompte tous = new TableauCompte(50);
    Banque(String n){
        nom = n;
    }
}
class Titulaire{
    String nom;
    Titulaire(String n){
        nom = n;
    }
    TableauCompte mesComptes = new TableauCompte(10);
}
class Compte{
    int numero;
    int solde;
    void depot(int n){
        solde = solde + n;
    }
    void retrait(int n){
        solde = solde - n;
    }
    void afficher(){
        Terminal.ecrireString("solde du compte numero " + numero + ": ");
        Terminal.ecrireInt(solde);
    }
}
class NonInitialise extends Exception{}
```

Question 17.2.1 - Account creation

Add to this skeleton class some methods. Only the bank can create new accounts. When it creates a new account, it adds it to the list of accounts of each holder of the account and it also adds to its own list of accounts. For example, if bank **b** creates account **c** for holders **t1** and **t2**, the same object **c** will be added to three tables: **b.tous**, **t1.mesComptes,** and **t2.mesComptes**.

In each of these three arrays, there will be **the same address**, the object **c**.

To achieve this account creation we need a method by which Bank creates an account with a parameter **list of holders** (eg; array form). It can also be helpful to write a method to add a new account for a holder.

Question 17.2.2 - Variables

Write a `main`() method that performs a situation where three holders, Paul, Peter and Fatima have accounts in bank BNP. Paul and Fatima have a joint account. Fatima has one more personal account, as well as Peter.

Draw the situation on a diagram showing all objects and arrays.

Question 17.2.3 - Transfers

We want to write a method that allows to make a transfer from an account held by the holder to an account that we only know the number (for which we are not necessarily the holder) and the bank.

In which class should we add this method?

Write this method.