

ArrayList

Question 1 *une matière*

On veut enregistrer dans un objet de la classe `ArrayList` les notes obtenues par un élève à une matière.

Le programme devra comporter les opérations suivantes, chacune réalisée par une méthode prenant en paramètre un objet de la classe `ArrayList` contenant les notes : ajouter une nouvelle note, afficher toutes les notes, calculer la moyenne. Le programme principal devra réaliser un test des méthodes avec les notes 12, 14, 9, un calcul de ma moyenne, puis ajout de la note 13 et nouveau calcul de la moyenne et affichage des notes.

```
import java.util.ArrayList;
public class Notes{
    public static void ajouter(ArrayList<Double> notes, Double lanote){
        notes.add(lanote);
    }
    public static Double moyenne(ArrayList<Double> notes){
        Double cumul = 0.0;
        for (int i=0; i<notes.size(); i++){
            cumul=cumul+notes.get(i);
        }
        return cumul/notes.size();
    }
    public static void afficher(ArrayList<Double> notes){
        Terminal.ecrireString("Notes:␣");
        for (int i=0; i<notes.size(); i++){
            Terminal.ecrireDouble(notes.get(i));
            Terminal.ecrireString("␣");
        }
        Terminal.sautDeLigne();
    }
    public static void main(String[] args){
        ArrayList<Double> lesnotes = new ArrayList<Double>();
        ajouter(lesnotes,12.0);
        ajouter(lesnotes,14.0);
        ajouter(lesnotes,9.0);
        afficher(lesnotes);
        Terminal.ecrireStringln("Moyenne:␣" + moyenne(lesnotes));
        ajouter(lesnotes,13.0);
        afficher(lesnotes);
        Terminal.ecrireStringln("Moyenne:␣" + moyenne(lesnotes));
    }
}
```

L'opération `ajouter` modifie le tableau donné en paramètre. Il n'est pas nécessaire de renvoyer un résultat (déclaration `void`). Le calcul de la moyenne doit renvoyer le résultat et non pas l'afficher. A noter que le type `double` doit s'écrire ici avec une majuscule.

Question 2 *correction des erreurs*

Une nouvelle méthode permettra de changer une note du tableau. Cette méthode peut être utile en cas d'erreur dans la saisie ou dans la correction de la copie.

Question 3 *menu*

Modifiez le programme principal pour que les différentes opérations soient proposées au moyen d'un menu.

```
import java.util.ArrayList;
public class Notes2{
    public static void ajouter(ArrayList<Double> notes, Double lanote){
        notes.add(lanote);
    }
    public static Double moyenne(ArrayList<Double> notes){
        Double cumul = 0.0;
        for (int i=0; i<notes.size(); i++){
            cumul=cumul+notes.get(i);
        }
        return cumul/notes.size();
    }
    public static void afficher(ArrayList<Double> notes){
        Terminal.ecrireString("Notes:");
        for (int i=0; i<notes.size(); i++){
            Terminal.ecrireDouble(notes.get(i));
            Terminal.ecrireString(" ");
        }
        Terminal.sautDeLigne();
    }
    public static void modifier(ArrayList<Double> notes, int ind,
                                Double lanote){
        notes.set(ind,lanote);
    }
    public static void saisieModif(ArrayList<Double> notes){
        int achanger;
        double nouvelle;
        for (int i=0; i<notes.size(); i++){
            Terminal.ecrireString("Note_\" + (i+1) + \":");
            Terminal.ecrireDoubleln(notes.get(i));
        }
        Terminal.ecrireString("Entrez le numéro de la note à changer:");
        achanger = Terminal.lireInt()-1;
        Terminal.ecrireString("Entrez la note correcte:");
        nouvelle = Terminal.lireDouble();
        modifier(notes,achanger,nouvelle);
    }
}
```

```

static void AfficheMenu (){
    String [] M ={"Afficher", "Ajouter", "Moyenne", "Modifier", "Quitter"};
    Terminal.ecrireStringln("Menu");
    for (int i=0; i<M.length;i++){
        Terminal.ecrireStringln((i+1)+"."+M[i]);
    }
    Terminal.ecrireString("Votre_choix:");
}
public static void main(String[] args){
    ArrayList<Double> lesnotes = new ArrayList<Double>();
    int rep;
    do{
        AfficheMenu();
        rep = Terminal.lireInt();
        if (rep == 1){
            afficher(lesnotes);
        } else if (rep == 2){
            Terminal.ecrireString("Entrez_la_note:");
            ajouter(lesnotes,Terminal.lireDouble());
        } else if (rep == 3){
            Terminal.ecrireStringln("Moyenne:" + moyenne(lesnotes));
        } else if (rep == 4){
            saisieModif(lesnotes);
        } else if (rep != 5){
            Terminal.ecrireStringln("Vous_devez_choisir_une_option");
            Terminal.ecrireStringln("(entier_compris_entre_1_et_5).");
        }
    } while(rep !=5);
}
}

```

Question 4 coefficients

A présent, on va doter les notes d'un certain coefficient entier à prendre en compte dans le calcul de la moyenne. Pour ce faire, on va enregistrer les données dans deux tableaux coordonnés : un tableau de notes et un tableau de coefficients. Le lien entre les deux se fera au moyen de l'indice : la note placée dans la case numéro *i* du tableau des notes aura le coefficient stocké dans la case numéro *i* du tableau des coefficients.

```

import java.util.ArrayList;
public class NotesCoeff{
    public static void ajouter(ArrayList<Double> notes, ArrayList<Integer>
                                coeffs, Double lanote, int coeff){
        notes.add(lanote);
        coeffs.add(coeff);
    }
    public static Double moyenne(ArrayList<Double> notes, ArrayList<Integer>
                                coeffs){
        Double cumul = 0.0;
        int totalcoeffs = 0;
        for (int i=0; i<notes.size(); i++){
            cumul=cumul+(notes.get(i)*coeffs.get(i));
        }
    }
}

```

```

        totalcoeffs = totalcoeffs + coeffs.get(i);
    }
    return cumul/totalcoeffs;
}
public static void afficher(ArrayList<Double> notes, ArrayList<Integer>
                           coeffs){
    Terminal.ecrireStringln("Notes:");
    for (int i=0; i<notes.size(); i++){
        Terminal.ecrireDouble(notes.get(i));
        Terminal.ecrireString(" _coefficient_");
        Terminal.ecrireIntln(coeffs.get(i));
    }
}
public static void modifier(ArrayList<Double> notes, int ind,
                           Double lanote){
    notes.set(ind,lanote);
}
public static void saisieModif(ArrayList<Double> notes){
    int achanger;
    double nouvelle;
    for (int i=0; i<notes.size(); i++){
        Terminal.ecrireString("Note _numero_ " + (i+1) + " :");
        Terminal.ecrireDoubleln(notes.get(i));
    }
    Terminal.ecrireString("Entrez _le _numéro _de _la _note _à _changer:");
    achanger = Terminal.lireInt()-1;
    Terminal.ecrireString("Entrez _la _note _correcte:");
    nouvelle = Terminal.lireDouble();
    modifier(notes,achanger,nouvelle);
}
static void AfficheMenu (){
    String [] M={"Afficher", "Ajouter", "Moyenne", "Modifier", "Quitter"};
    Terminal.ecrireStringln("Menu");
    for (int i=0; i<M.length;i++){
        Terminal.ecrireStringln((i+1)+"."+M[i]);
    }
    Terminal.ecrireString("Votre _choix:");
}
public static void main(String[] args){
    ArrayList<Double> lesnotes = new ArrayList<Double>();
    ArrayList<Integer> lescoeffs = new ArrayList<Integer>();
    int rep;
    double note;
    int coeff;
    do{
        AfficheMenu();
        rep = Terminal.lireInt();
        if (rep == 1){
            afficher(lesnotes,lescoeffs);
        }else if (rep == 2){
            Terminal.ecrireString("Entrez _la _note:");
            note = Terminal.lireDouble();
            Terminal.ecrireString("Entrez _le _coefficient:");

```

```

        coeff = Terminal.lireInt();
        ajouter(lesnotes,lescoeffs,note,coeff);
    }else if (rep == 3){
        Terminal.ecrireStringln("Moyenne:_" +
                                moyenne(lesnotes,lescoeffs));
    }else if (rep == 4){
        saisieModif(lesnotes);
    }else if (rep != 5){
        Terminal.ecrireStringln("Vous_devez_choisir_une_option");
        Terminal.ecrireStringln("(entier_compris_entre_1_et_5).");
    }
    }while(rep !=5);
}
}

```

Exercice 6.2.1 ensembles

Question 1 création d'ensembles

On va utiliser des objets de la classe `ArrayList` pour représenter des ensembles de nombres entiers. Le programme proposera les opérations suivantes :

- créer un singleton (ensemble à un élément, donné en paramètre). Cette opération crée un nouvel ensemble, donc un nouvel objet de la classe `ArrayList`.
- ajouter un élément à un ensemble existant. Celui-ci sera donc modifié.
- afficher un ensemble.
- calculer l'union de deux ensembles. Le résultat est un nouvel ensemble, donc un nouvel objet de la classe `ArrayList`.
- calculer l'intersection de deux ensembles. Le résultat est un nouvel ensemble, donc un nouvel objet de la classe `ArrayList`.

Le programme principal fera quelques appels aux différentes méthodes.

```

import java.util.ArrayList;
public class Ens{
    public static ArrayList<Integer> singleton(int x){
        ArrayList<Integer> res = new ArrayList<Integer>();
        res.add(x);
        return res;
    }
    public static void ajouter(ArrayList<Integer> e, int x){
        if (! e.contains(x)){
            e.add(x);
        }
    }
    public static void afficher(ArrayList<Integer> e){
        Terminal.ecrireChar(' ');
        for (int i=0; i<e.size()-1;i++){
            Terminal.ecrireInt(e.get(i));
            Terminal.ecrireChar(',');
        }
        if (e.size()>0){

```

```

        Terminal.ecrireInt(e.get(e.size()-1));
    }
    Terminal.ecrireChar(' ');
}
public static ArrayList<Integer> union(ArrayList<Integer> e1,
                                     ArrayList<Integer> e2){
    ArrayList<Integer> res = new ArrayList<Integer>();
    for (int i=0; i<e1.size(); i++){
        ajouter(res,e1.get(i));
    }
    for (int i=0; i<e2.size(); i++){
        ajouter(res,e2.get(i));
    }
    return res;
}
public static ArrayList<Integer> inter(ArrayList<Integer> e1,
                                     ArrayList<Integer> e2){
    ArrayList<Integer> res = new ArrayList<Integer>();
    for (int i=0; i<e1.size(); i++){
        if (e2.contains(e1.get(i))){
            ajouter(res,e1.get(i));
        }
    }
    return res;
}
public static void main(String[] args){
    ArrayList<Integer> ens1, ens2, ens3, ens4;
    ens1 = singleton(1);
    Terminal.ecrireString("ens1=");
    afficher(ens1);
    Terminal.sautDeLigne();
    ens2=singleton(4);
    ajouter(ens2,5);
    Terminal.ecrireString("ens2=");
    afficher(ens2);
    Terminal.sautDeLigne();
    ens3=union(ens1,ens2);
    Terminal.ecrireString("ens3=");
    afficher(ens3);
    Terminal.sautDeLigne();
    ajouter(ens3,8);
    Terminal.ecrireString("ens3=");
    afficher(ens3);
    Terminal.sautDeLigne();
    ens4=inter(ens3,ens2);
    Terminal.ecrireString("ens4=");
    afficher(ens4);
    Terminal.sautDeLigne();
}
}

```

Question 2 propriétés des ensembles

Ajouter les deux opérations suivantes :

- tester l'appartenance d'un élément à un ensemble.
- donner le nombre d'éléments dans un ensemble.

```
import java.util.ArrayList;
public class Ens2{
    public static boolean appartient(ArrayList<Integer> e, int x){
        return e.contains(x);
    }
    public static int cardinal(ArrayList<Integer> e){
        return e.size();
    }
    public static ArrayList<Integer> singleton(int x){
        ArrayList<Integer> res = new ArrayList<Integer>();
        res.add(x);
        return res;
    }
    public static void ajouter(ArrayList<Integer> e, int x){
        if (! e.contains(x)){
            e.add(x);
        }
    }
    public static void afficher(ArrayList<Integer> e){
        Terminal.ecrireChar(' ');
        for (int i=0; i<e.size()-1;i++){
            Terminal.ecrireInt(e.get(i));
            Terminal.ecrireChar(', ');
        }
        if (e.size()>0){
            Terminal.ecrireInt(e.get(e.size()-1));
        }
        Terminal.ecrireChar(' ');
    }
    public static ArrayList<Integer> union(ArrayList<Integer> e1,
                                           ArrayList<Integer> e2){
        ArrayList<Integer> res = new ArrayList<Integer>();
        for (int i=0; i<e1.size(); i++){
            ajouter(res,e1.get(i));
        }
        for (int i=0; i<e2.size(); i++){
            ajouter(res,e2.get(i));
        }
        return res;
    }
    public static ArrayList<Integer> inter(ArrayList<Integer> e1,
                                           ArrayList<Integer> e2){
        ArrayList<Integer> res = new ArrayList<Integer>();
        for (int i=0; i<e1.size(); i++){
            if (e2.contains(e1.get(i))){
                ajouter(res,e1.get(i));
            }
        }
    }
}
```

```
    }  
    return res;  
}  
public static void main(String[] args){  
    ArrayList<Integer> ens1, ens2, ens3, ens4;  
    ens1 = singleton(1);  
    Terminal.ecrireString("ens1=");  
    afficher(ens1);  
    Terminal.sautDeLigne();  
    ens2=singleton(4);  
    ajouter(ens2,5);  
    ajouter(ens2,3);  
    ajouter(ens2,5);  
    ajouter(ens2,3);  
    Terminal.ecrireString("ens2=");  
    afficher(ens2);  
    Terminal.sautDeLigne();  
    Terminal.ecrireString("Card(ens2)=");  
    Terminal.ecrireInt(cardinal(ens2));  
    Terminal.sautDeLigne();  
    Terminal.ecrireStringln("3_appartient_a_ens2=" + appartient(ens2,3));  
    Terminal.ecrireStringln("9_appartient_a_ens2=" + appartient(ens2,9));  
}
```
