

Exercices sur les classes (suite)

Exercice 2.2.1 *comptes (suite)*

On veut améliorer et compléter la classe `Compte` suivante :

Listing 2.1 – (pas de lien)

```
class Compte{
    int solde = 0;
    void deposer(int montant){
        solde = solde + montant;
    }
    void retirer(int montant){
        solde = solde -montant;
    }
    void virerVers(int montant, Compte destination){
        destination.retirer(montant);
        this.deposer(montant);
    }
    void afficher(){
        Terminal.ecrireString("solde:␣"+ solde);
    }
}
```

1. modifier la méthode `retirer` pour empêcher le retrait quand le compte n'est pas suffisamment approvisionné.
2. modifier la classe pour ajouter à chaque compte un numéro de compte.

Exercice 2.2.2 *bibliothèque*

On veut écrire un programme qui représente une bibliothèque avec des livres.

Question 1

Ecrivez une classe permettant de représenter des livres avec le titre, le nom de l'auteur et l'éditeur. Les opérations consisteront en saisir un livre, ce qui crée un nouvel objet et afficher les informations d'un objet `livre`. Vous ferez attention pour chaque opération de déterminer si ce doit être une méthode statique ou non.

Question 2

Une bibliothèque est une collection de plusieurs livres. Les livres seront représentés au moyen de la classe écrite pour répondre à la question 1. Ecrivez une classe bibliothèque permettant de stocker plusieurs livres, avec deux opérations : ajouter un livre, afficher le contenu de la bibliothèque. Vous écrirez une méthode `main` qui crée un ou deux livres, une bibliothèque et qui appelle les différentes méthodes disponibles.

Exercice 2.2.3 *tableau trié*

Nous allons voir dans cet exercice comment on peut avoir un tableau trié sans jamais faire une opération de tri.

On va utiliser des produits comportant un nom, une référence et un prix hors taxe. La référence sera un nombre entier. On veut écrire une classe `Stock` qui contient une liste de produits représentée par un tableau de produits trié par ordre croissant de la référence.

Un stock sera initialement vide, puis rempli petit à petit au moyen d'une opération d'ajout. Ce sera à cette opération d'ajout d'insérer le nouvel objet à la bonne place du tableau, compte tenu de l'ordre des références.

Outre cette opération d'ajout, vous ferez une opération qui affiche le contenu d'un stock et une opération qui recherche le prix d'une référence donnée dans un stock. Vous structurerez au mieux les deux classes `Produit` et `Stock` pour que chacune d'elle comporte les méthodes concernant leurs données respectives. Vous mettrez la méthode `main` dans une troisième classe.

Exercice 2.2.4 *polynômes*

Un polynôme est une fonction qui à une variable numérique x associe la somme de facteurs obtenus en multipliant une puissance de x par un certain coefficient entier. Par exemple $f(x) = 3 * x^3 - 5 * x^2 + 2$. Noter que le coefficient $+2$ s'applique implicitement à la puissance x^0 , qui vaut 1.

L'addition de polynômes se fait en additionnant les coefficients puissance par puissance. Par exemple $(3 * x^3 - 5 * x^2 + 2) + (2 * x^3 - 4 * x - 3) = 5 * x^3 - 5 * x^2 - 4 * x - 1$

La multiplication de polynômes se fait en utilisant les propriétés de distributivité de la multiplication par rapport à l'addition. Par exemple

$$\begin{aligned} (3 * x^3 - 5 * x^2 + 2) * (2 * x^3 - 4 * x - 3) &= (3 * x^3) * (2 * x^3 - 4 * x - 3) + (-5 * x^2) * (2 * x^3 - 4 * x - 3) + 2 * (2 * x^3 - 4 * x - 3) \\ &= (6 * x^6 - 12 * x^4 - 9 * x^3) + (-10 * x^5 + 20 * x^3 + 15 * x^2) + (4 * x^3 - 8 * x - 6) \\ &= 6 * x^6 - 10 * x^5 - 12 * x^4 + 15 * x^3 + 15 * x^2 - 8 * x - 6 \end{aligned}$$

Ecrivez une classe des polynômes avec les opérations d'addition, de multiplication et de calcul de la valeur du polynôme pour un x donné.