

3. Feuille de TP 10

BUT 1A Informatique 2022 - 2023

S.ANGLADE

semaine 10 : TP

Object

1. Dans l'exécutable suivant, ajouter un morceau de code pour calculer la somme des éléments de `tableau` sous la forme d'un entier. Vous noterez que la classe *Number* est la classe ancêtre de tous les nombres et permet les conversion dans les types de base tels que `int`, `float`, ..., via des méthodes ad hoc (`doubleValue()`, `floatValue()`, ...).

```
public class ExecNumber
{
    public static void main(String [] args)
    {
        List<Number> tableau = new ArrayList<>();
        tableau.add(5);
        tableau.add(6.);
        tableau.add(7.f);
        Number number = 8.5f;
        tableau.add(number);
    }
}
```

2. Quelles sont les méthodes `toString()` qui sont appelées lorsqu'on exécute l'instruction suivante :

```
System.out.println(tableau);
```

3. Quel sera l'affichage généré par l'exécution du code suivant ? Pourquoi ?

```
Number x = new Integer(5);
Number y = new Double(5);
System.out.println(x.equals(y));
```

Notez le message de warning affiché, lié à la dépréciation de ces écritures.

Magasins

1. Reprendre la classe *Magasin* que vous avez définie dans le TP2 de la première période (ouvert le lundi et/ou le dimanche).
2. On vous propose cette fois-ci de modéliser les magasins à l'aide d'un ensemble : vous devez pouvoir ajouter un magasin à votre ensemble, afficher tous vos magasins, ...
3. Déterminez l'ensemble des magasins qui sont ouverts le lundi.
4. Donner un exécutable permettant de tester vos diverses classes et méthodes.

Produit / Achats

1. On vous demande de définir une classe **Produit** dont les attributs sont un nom et un prix unitaire.

```
public class Executable{
    public static void main(String[] args)
    {
        Produit salade1 = new Produit("Salade", 1.5);
        Produit brocolis = new Produit("Brocolis", 2.75);
        System.out.println(salade1);
        // Salade pour 1.50 euro(s).
    }
}
```

2. Définir une classe **Achats** représentée par un dictionnaire de clef Produit et de valeur quantité de produit achetée.

```
public class Executable{
    public static void main(String[] args)
    {
        Achats achats = new Achats();
        //
        // Ajouts d'achats (à voir dans la suite)
        //
        System.out.println(achats);
        // 2 Salades pour 3 euros
        // 1 Baguette pour 1 euro(s)
    }
}
```

1. Vous devez pouvoir ajouter à vos achats un produit en une certaine quantité. Dans le cas où le produit serait déjà présent parmi vos achats, n'oubliez pas de modifier la quantité.
2. On veut connaître la facture totale de vos achats.
3. On voudrait trier les achats selon deux critères : soit par nom de produit, soit par prix unitaire croissant, définir une méthode pour cela.

Occurrences

Définir une classe **Occurrences** qui va créer le dictionnaire des occurrences d'entiers d'une liste d'entiers. Si votre liste contient les valeurs `[1, 5, -3, 8, 2, -3, 5, -3]` les couples (clé, valeur) de votre dictionnaire seront `(1, 1) (5, 2) (-3, 3) (8, 1) (2, 1)`.

Recettes et Magasins

1. On vous demande de définir une classe **Recette** permettant de créer une recette du type ("oeufs", 3) ("lait", 0.5) ("farine", 0.250) pour la recette de crêpes par exemple. Vous devrez prévoir une méthode **ajouteIngredient** permettant d'ajouter à une recette un ingrédient (son nom) et la quantité nécessaire. Quelle structure de données allez-vous utiliser ? Expliquez.
2. Définir maintenant une classe **Magasin** permettant de connaître les ingrédients et leur prix unitaire, présents dans un magasin donné. Vous devrez prévoir une méthode **ajouterIngredient** qui ajoute au magasin le couple (nom, prix unitaire). Quelle structure de données allez vous utiliser ? Expliquez.
3. Définissez une classe **RecetteMagasin** contenant deux méthodes : la première nommée *recettePossible* qui prend en paramètre une recette et un magasin et vérifie que les ingrédients de la recette peuvent être achetés

dans ce magasin. La seconde *prixRecette* a comme paramètres une recette et un magasin et retourne le prix total de la recette dans ce magasin.

4. Testez toutes ces classes dans un exécutable.