

Devoir

1. Écrivez une classe `Item`. Un article est identifié par un nom sous forme de chaîne de caractères, un prix (flottant) et enfin un identifiant unique (entier positive ≥ 1). On peut également donner une description à l'objet mais c'est optionnel. Le code suivant vous montre la création de deux articles :

```
public static void main ( String [] args) {  
    Item i1 = new Item ( "Vodka" , "Une bouteille de Vodka" , 5483918746738L, 75.5);  
    Item i2 = new Item ( "Filet de citron 200g" , 5473664615361L, 3.25);  
}
```

Ajoutez également deux méthodes `getPrice` a la classe :

- `public double getPrice();`
- `public double getPrice (int q);`

La première méthode renvoie simplement le prix de l'article, tandis que la seconde renvoie le prix pour `q` fois l'article en tenant compte du fait que si on commande plus de 5 fois un article, on reçoit 5% de réduction sur le prix total.

Ecrivez aussi une classe `TestItem` afin de tester la classe `Item`. Attention : deux items ne peuvent avoir le même identifiant!

Devoir

2. Mettez en œuvre la classe **Produit**. Un produit a:

- Code unique (integer)
- Prix (double)
- La date de production (GregorianCalendar "jj / mm / aa")
- Période de validité - en jours (integer)
- Catégorie (string; EX.: «aliment»)

Mettez en œuvre les constructeurs et les méthodes nécessaires. Chaque **boutique** peut vendre jusqu'à 50 produits. Une boutique contient une gamme de produits et les méthodes suivantes:

- Ajoutez un nouveau produit
- Supprimez un produit en utilisant son index dans le tableau
- Modifiez le prix d'un produit, étant donné le code unique
- Affichez tous les produits à partir d'une certaine catégorie
- Affichez tous les produits qui sont périmés
- Affichez le produit le moins cher
- Calculez la moyenne des prix des produits fabriqués au cours des trois dernières années

Devoir

Créez une classe de test et de déclarez 2 boutiques.

Ajoutez 6 produits de la même catégorie pour la première boutique: 3 produits sont périmés et 3 sont valables. Ajoutez encore 2 produits des différentes catégories pour la première boutique. Testez ensuite les méthodes.

```
Livre l1 = new Livre();
```

```
l1.setDatePublication(new GregorianCalendar(2012,12,29));
```

Pour afficher dans un format agréable:

```
SimpleDateFormat formatter = new SimpleDateFormat("dd/MM/yy");
```

```
String dp = formatter.format(datePublication.getTime());
```

Note: Si vous ne réussissez pas à travailler avec **GregorianCalendar**, vous pouvez mettre la date de production seulement en années