

Université d'Aix-Marseille – L2 Informatique  
Introduction à la programmation orientée objet  
TP *n*°2

## Classes et instances

1. Écrire la classe **Verre** ayant les deux attributs entiers **contenance** et **quantite**, un constructeur prenant en paramètre la contenance (en cl) du verre créé, la méthode **void emplir(int q)** qui ajoute **q** cl de liquide à la quantité contenue dans le verre et la méthode **boire(int q)** qui fait l'inverse.
2. Écrire la classe **Bouteille** ayant un attribut entier **quantite** et un attribut booléen **estouverte**, un constructeur prenant en paramètre la quantité initiale (en cl) de la bouteille créée, les méthodes **void ouvrir()** et **void fermer()** et la méthode **void verseDans(Verre v, int q)** qui verse la quantité **q** de liquide de la bouteille dans le verre **v**.
3. Ajouter une méthode **main** à la classe **Bouteille** pour exécuter la suite d'opérations suivante : création d'une bouteille de jus d'orange et d'une bouteille de jus de citron d'un litre chacune, création d'un verre de contenance 20cl, versement de 8cl de citron puis de 12cl d'orange dans le verre, puis consommation de tout le verre.

## Composition de classes

On veut modéliser des bâtiments composés de bureaux, où des employés peuvent se déplacer d'un bureau à l'autre, d'un bâtiment à l'autre. Pour ceci, on va créer les classes **Bureau**, **Batiment** et **Employe**.

1. Écrire la classe **Bureau** qui possède un numéro et qui est ouvert ou fermé. Ses méthodes sont :
  - un constructeur qui prend en paramètre le numéro du bureau.
  - la méthode **boolean ouvert()** qui indique si le bureau est ouvert ou fermé (fermé à sa création).
  - les méthodes **void ouvrir()** et **void fermer()** qui ouvre et ferme le bureau.
2. Écrire la classe **Batiment** qui possède un numéro et un certain nombre de bureaux. Elle contient un constructeur qui prend en paramètre le numéro du bâtiment et le nombre **n** de bureaux qu'il contient. Ses bureaux sont numérotés de 1 à **n**.
3. Écrire la classe **Employe** qui permet de mémoriser le bureau et le bâtiment dans lesquels un employé se trouve. Par ailleurs, on souhaite connaître le nombre total d'employés créés. Ses méthodes sont :
  - un constructeur, qui prend un bâtiment et un numéro de bureau en paramètre pour déterminer sa position initiale.
  - la méthode **void aller(Batiment t, int n)** qui fait se déplacer l'employé de sa position actuelle vers le bureau numéro **n** du bâtiment **t**. Si le bureau que quitte l'employé est fermé, il faut qu'il l'ouvre. Même chose pour le bureau dans lequel il rentre. La méthode affichera les étapes de son parcours : le bureau qu'il quitte, le fait qu'il l'ouvre (éventuellement), le fait qu'il change de bâtiment (si c'est le cas), le bureau dans lequel il arrive et s'il l'ouvre pour rentrer dedans.
4. Ajouter une méthode **main** à la classe **Batiment** pour faire un programme qui crée un bâtiment à 3 bureaux, un bâtiment à 4 bureaux, un employé qui se trouve initialement dans le bureau numéro 3 du premier bâtiment puis qui va dans le bureau numéro 2 du deuxième bâtiment.

L'ensemble des bâtiments existants doit être surveillé en dehors des heures de bureaux. Pour ceci, on installe un système de sécurité qui, lorsqu'on le déclenche, ferme automatiquement tous les bureaux de tous les bâtiments existants. Dès lors, si un bureau est ouvert, une alarme se déclenche.

5. Ajouter des attributs de classes, modifier les méthodes des classes précédemment écrites et ajouter la méthode de classe **static void metAlarme(boolean b)** (qui démarre ou stoppe le système de sécurité) de manière à obtenir un comportement des classes tel que décrit ci-dessus.
6. Modifier la méthode **main** en ajoutant un déclenchement de système de sécurité puis un déplacement de l'employé.