

Un laboratoire pharmaceutique fabrique et distribue des échantillons de médicament auprès de médecins dans l'optique d'une plus grande prescription de leurs articles. Afin d'optimiser la gestion de ses stocks d'échantillon de médicament, le laboratoire souhaite se munir d'une application Java que vous êtes chargés d'initier.

Un médicament est identifié par un nom, un effet et une posologie. On enregistre également deux prix : le prix de production de l'échantillon et le prix de vente.

1/ Créer la classe Medicament (cette classe sera enrichie tout au long du TP).

2/ Ecrire un constructeur sans argument et un second constructeur permettant d'initialiser tous les attributs de l'objet à l'instanciation.

3/ Ajouter des accesseurs (*getters*) de manière à pouvoir obtenir l'information contenue dans les attributs de l'objet.

4/ Ajouter des mutateurs (*setters*) de manière à pouvoir modifier les prix du médicament ou mettre à jour l'effet ou la posologie. Attention, une modification ne doit pas mettre un objet médicament dans un état invalide (e.g. prix de production négatif ou prix de vente inférieur au prix de production).

5/ Construire un programme de test pour vérifier le bon fonctionnement de la classe Medicament. Créer une classe TestAppli qui contiendra une méthode main pour réaliser les opérations suivantes :

• Création des objets

Nom médicament	Effet	Posologie	Prix de l'échantillon	Prix de vente
Doliprane	soulage les maux de tête	1 comprimé toutes les 6 heures	2.5	2.8
Strepsil	soulage les maux de gorge	1 pastille toutes les 2 heures	3.5	4.2
Arnica	Apaise les douleurs de coups et bosses	Appliquer sur le choc	4.2	5.5

• Manipulation des objets :

- Afficher le nom, la posologie et l'effet du doliprane
- Afficher le prix d'échantillon et le prix de vente du strepsil
- Modifier le prix d'échantillon et le prix de vente de l'arnica, respectivement 3.8 et 5.2

6/ Ajouter une méthode qui permet d'afficher toutes les informations d'un médicament.

7/ Ajouter une méthode qui permet d'afficher une vision synthétique du médicament : nom, effet et prix de vente.

8/ Le laboratoire souhaite connaître le montant de la marge gagnée sur la vente de chaque médicament ainsi que le pourcentage que la marge représente par rapport au prix de l'échantillon. Ajouter les méthodes nécessaires pour calculer ces informations.

9/ Enrichir la classe de test pour tester les méthodes implémentées dans le cadre des tâches 6 à 8.

4/ Ajouter dans la classe de test : TestAppli du TP1 la création des objets de la classe StockMed afin d'enregistrer les informations suivantes :

médicament	Quantité en stock	Seuil de commande	Quantité commandée
doliprane	150	100	200
strepsil	100	90	50
arnica	250	120	300

5/Tester les méthodes de la classe StockMed afin de réaliser les opérations suivantes :

- 25 boîtes de Strepsil, 100 boîtes de Doliprane et 100 boîtes d'Arnica sont sorties du stock.
- le laboratoire a reçu une livraison de 50 boîtes de Strepsil et 200 boîtes de Doliprane.
- le laboratoire veut connaître le prix total de son stock de médicaments.



## Module M2103 – Java

### TP Gestion des stocks – Ecriture de la classe StockMed liée à la classe Medicament

Pour gérer les échantillons de médicaments stockés par le laboratoire avant d'être distribués aux médecins, on utilise la méthode du point de commande. Ce principe consiste à commander la production d'une certaine quantité de boîtes dès que la quantité en stock atteint un seuil minimum de stock, appelé point de commande. La quantité commandée permettra de revenir à une quantité donnée appelée stock complet.

Créer la classe StockMed à partir des informations suivantes :

1/ La classe StockMed doit contenir les informations d'un stock de médicament : médicament stocké (de type Medicament), quantité en stock, seuil de commande, stock complet

2/ Constructeurs de la classe :

- Un constructeur sans argument qui construit un médicament vide, initialise toutes les valeurs de chaîne à vides et les valeurs numériques à 0
- Un constructeur avec un paramètre pour chaque attribut de la classe qui initialise les valeurs des attributs avec les valeurs passées en paramètre.

3/ Créer les méthodes permettant de réaliser les opérations suivantes :

- Afficher le détail du stock sous le modèle suivant :  
[codeMédicament] [NomMédicament ]  
Effets : [Effet du médicament]  
Posologie : [Posologie du médicament]  
Prix de l'échantillon : [Prix Echantillon]  
Prix de vente : [Prix Vente]  
Quantité en Stock : [quantité en stock]
- Ajouter des médicaments au stock suite à une livraison  
Cette fonction permet de modifier la quantité de médicament en stock en ajoutant une quantité qui aura été livrée. Cette méthode prend un paramètre de type entier contenant la quantité livrée.
- Supprimer des médicaments du stock suite à une commande  
Cette fonction permet de modifier la quantité de médicament en stock en enlevant une quantité qui aura été commandée par un client à condition qu'on n'atteigne pas un stock négatif. Cette méthode affiche le message « Commande de [quantité commandée] déclenchée » lorsque la quantité en stock se trouve en dessous du seuil de commande et le message « Commande refusée » si le stock n'est pas suffisant. Cette méthode prend un paramètre de type entier contenant la quantité commandée.
- Calculer le prix total du stock pour le laboratoire (basé sur le prix de l'échantillon)