

Objectifs :

- ✓ Les classes abstraites
- ✓ polymorphisme via les classes abstraites

On souhaite gérer le salaire des employés d'une société. Un employé est caractérisé par son nom, son prénom, son âge, son statut et sa date d'embauche.

L'objectif est de coder une classe abstraite « **Employe** » dotée des attributs nécessaires, d'une méthode abstraite « **CalSalaire** » qui calcule le salaire d'un employé (ce calcul dépendra en effet du type de l'employé). Une méthode « **getAge** » qui retourne l'âge du salarié, La classe contient aussi une méthode « **getNom** » retournant une chaîne de caractère obtenue en concaténant la chaîne de caractères "l'employé:" avec le prénom et le nom. Exemple :« l'employé: said Hamadi »
N'oubliez pas un constructeur prenant en paramètre l'ensemble des attributs nécessaires.

Partie I : Calcul du salaire

Le calcul du salaire mensuel dépend du statut de l'employé. On distingue deux types des employés (Les Vendeurs et les agents de production)

Il existe les vendeurs affectés à la **vente en ligne** et les autres à la **vente directe** des produits :

- Le salaire mensuel des vendeurs en ligne est le 30 % du *chiffre d'affaire* qu'ils réalisent mensuellement, plus un bonus de 600dh. Le salaire des vendeurs en directe est également le 25 % du *chiffre d'affaire* qu'ils réalisent mensuellement plus un bonus de 750DH.

Les agents de production eux aussi sont trois types, les **chefs de projets**, les **ingénieurs** et les **techniciens**.

Le salaire des chefs vaut le nombre de projets réalisés multipliés par 5000 plus un bonus de 400DH, quand aux ingénieurs, leur salaire vaut le nombre d'heures de travail mensuel multipliés par 300dh et le salaire des techniciens est le nombre d'unités produites multipliés par 70.

Codez en JAVA une hiérarchie de classes pour les employés en respectant les conditions suivantes :

- La super-classe de la hiérarchie doit être la classe « **Employe** ».
- Les sous-classes doivent contenir les attributs qui leur sont spécifiques ainsi que le codage approprié des méthodes « **CalSalaire** » et **getNom**, en changeant le mot "l'employé" par la catégorie correspondante et « **getAge** ».
- Chaque sous-classe est dotée de constructeur prenant en argument l'ensemble des attributs nécessaires.
- N'hésitez pas à introduire des classes intermédiaires pour éviter au maximum les redondances d'attributs et de méthodes dans les sous-classes.

Partie II : Calcul des primes

Nous voulons donner des primes mensuelles aux employés de types techniciens et vendeurs:

- Pour les vendeurs, leur prime est calculée selon le chiffre d'affaire (CA) réalisé, si $CA > 50000dh$ la prime est 2% du salaire dh sinon si $CA > 100000dh$ la prime sera 5% du salaire
- Pour les techniciens, la prime est calculée selon le nombre d'heures travaillées mensuel si le nombre d'heures > 240 la prime vaut 2% du salaire.
- **Ajouter une interface « Salarie » contenant les méthodes de calcul des primes selon le type de l'employé (Vendeur ou Technicien). Le salaire est entré comme paramètre des méthodes)**
- **Implémente l'interface « Salarie ».**
- **Dans une classe principale, tester toutes les méthodes**