

Laboratoire 6

Les classes Employe et Entreprise

La classe Employe

Dans un premier temps, nous allons écrire une classe nommée **Employe**. Un employé est caractérisé par son code, son nom (chaîne), son prénom (chaîne), son salaire (réel), son année d'embauche (entier), son adresse courriel (chaîne), son login (chaîne) et son mot de passe (chaîne). Vous devez prendre en considération les règles suivantes pour les attributs :

- Le code s'incrémente lors de la création d'un nouvel employé.
- Les noms sont en majuscules.
- Les prénoms sont en minuscules.
- Le salaire doit être supérieur à 1000\$. Sinon la valeur 1000 est affecté par défaut.
- L'adresse courriel est générée par la méthode nommée *creerEmail*.
- Le login est généré par la méthode nommée *creerLogin*.
- L'année d'embauche doit être supérieure à 1970. Sinon on affecte la valeur 1970 comme valeur par défaut.
- Le mot de passe doit être composé, au minimum, de six caractères (lettres ou chiffres). Sinon on affecte la chaîne « ABC963ABC » comme valeur par défaut.

On vous demande de :

- 1- Définir la classe **Employe** et ses attributs.
- 2- Définir les propriétés associées aux attributs.
- 3- Définir un constructeur initialisant le nom et le prénom.
- 4- Définir un constructeur initialisant le nom, le prénom, le salaire et l'année d'embauche.
- 5- Définir un constructeur initialisant le nom, le prénom, le salaire, l'année d'embauche et le mot de passe.
- 6- Définir une méthode nommée *CreerLogin* qui permet d'affecter une valeur à l'attribut login. Cette valeur est générée par la concaténation des deux premiers caractères du nom avec les 4 premiers caractères du prénom séparés par un point.

Exemple :

Nom = BOUHLEL	Deux premiers caractères :	BO
Prénom = mohamed	Quatre premiers caractères :	moha
Login	Concaténation	BO.moha

- 7- Définir une méthode nommée **CreerEmail** qui permet d'affecter une valeur à l'attribut représentant l'adresse courriel. Cette valeur est générée par la concaténation du prénom avec le premier caractère du nom séparer par un point. Toutes les adresses courriels se terminent par « @proIT.ca ».

Exemple :

Nom = BOUHLEL	Premier caractère	B
Prénom = mohamed	Le prénom	mohamed
Adresse courriel	Concaténation	mohamed.B@proIT.ca

- 8- Définir une méthode nommée **SalaireAnn** permettant de calculer le salaire annuel de l'employé courant. Le salaire annuel est calculé par la multiplication du salaire par 12.
- 9- Définir une méthode nommée **Anciennete** permettant de calculer le nombre d'année d'ancienneté de l'employé courant. Pour simplifier, calculer l'ancienneté par rapport à 2021.
- 10- Définir une méthode nommée **AugmentationSalaire** qui permet d'augmenter le salaire de l'employé selon son ancienneté en appliquant la formule suivante :
- Si l'ancienneté est inférieure à 3 ans alors on augmente de 5%.
 - Si l'ancienneté est supérieure ou égale à 3 ans et inférieure à 7 ans alors on augmente de 10%.
 - Si l'ancienneté est supérieure ou égale à 7 ans et inférieure à 15 ans alors on augmente de 15%.
 - Si l'ancienneté est supérieure ou égale à 15 ans alors on augmente de 20%.
- 11- Surcharger la méthode **AugmentationSalaire** pour augmenter le salaire d'un employé par un pourcentage fourni en paramètre.
- 12- Définir une méthode **ToString** permettant de convertir un employé vers une chaîne de caractères. La signature de la méthode est comme suit : `public String ToString()`. La représentation de l'employé est comme suit :

```
Code : 1
Nom : BOUHLEL mohamed
Année embauche : 1980
Courriel : mohamed.B@proIT.ca
Login : BO.moha
Salaire annuel : 160000
```

- 13- Définir une méthode nommée **Egalite** permettant de tester l'égalité de l'employé courant avec l'employé passé comme paramètre. Deux employés sont égaux s'ils ont le même code et le même nom de famille.

La classe Entreprise

Dans un second temps, nous allons écrire une classe nommée **Entreprise**. Une entreprise est caractérisée par un nom (string) et une liste d'employé (List<Employe>).

On vous demande de :

- 1- Définir la classe **Entreprise** avec ses attributs.
- 2- Définir un constructeur permettant d'initialiser une entreprise par son nom.

- 3- Définir une méthode nommée **AddEmploye** permettant d'ajouter un employé à l'entreprise.
- 4- Surcharger la méthode **AddEmploye** pour ajouter un employé à l'entreprise en donnant son nom et son prénom.
- 5- Surcharger la méthode **AddEmploye** pour ajouter un employé à l'entreprise en donnant son nom, son prénom, son salaire et son année d'embauche.
- 6- Surcharger la méthode **AddEmploye** pour ajouter un employé à l'entreprise en donnant son nom, son prénom, son salaire, son année d'embauche et son mot de passe.
- 7- Définir une méthode nommée **FindEmploye** permettant de chercher un employé dans l'entreprise en fournissant son code. La méthode retourne *null* si le code ne correspond à aucun employé sinon elle retourne l'employé en question.
- 8- Surcharger la méthode **FindEmploye** permettant de chercher un employé dans l'entreprise en fournissant son nom et son prénom. La méthode retourne *null* si le nom et le prénom ne correspondent à aucun employé. Sinon elle retourne le premier employé dont le nom et prénom ont été fournis.
- 9- Définir une méthode nommée **DelEmploye** permettant de supprimer un employé de l'entreprise en fournissant son code. La méthode retourne un booléen indiquant si l'opération de suppression a réussi ou non.
- 10- Surcharger la méthode **DelEmploye** pour supprimer un employé en fournissant son nom et son prénom. La méthode retourne un booléen indiquant si l'opération de suppression a réussi ou non.
- 11- Définir une méthode nommée **SalaireMensuel** permettant de calculer le salaire mensuel de tous les employés de l'entreprise.
- 12- Définir une méthode nommée **AugmentationSalaire** permettant d'augmenter le salaire de tous les employés de 15%.
- 13- Définir une méthode nommée **ChargeTotale** permettant de calculer la charge annuelle de l'entreprise. La charge annuelle est calculée par la formule suivante : salaire mensuel * 12.
- 14- Définir une méthode **ToString** permettant de convertir une entreprise vers une chaîne de caractères. La représentation de l'entreprise est libre

La classe TestEntreprise

Enfin, on vous demande d'écrire une classe nommée **TestEntreprise** pour tester les deux classes précédentes. Cette classe contient uniquement la méthode **Main** et doit réaliser les opérations suivantes :

- 1- Créer un employé nommé **e1** avec le premier constructeur de la classe Employe.
- 2- Créer un employé nommé **e2** avec le second constructeur de la classe Employe.
- 3- Créer deux employés nommés **e3** et **e4** avec le troisième constructeur de la classe Employe.
- 4- Affecter le salaire 5000 à l'employé **e1**.
- 5- Afficher les employés **e1** et **e2**.
- 6- Afficher l'adresse courriel de **e3** et le login de **e4**.
- 7- Calculer et afficher le salaire annuel de **e1** et **e3**.
- 8- Calculer et afficher l'ancienneté de **e1** et **e4**.
- 9- Augmenter et afficher le salaire de **e2** et de **e3** selon leur ancienneté.
- 10- Augmenter le salaire de **e4** de 35% et afficher la nouvelle valeur.
- 11- Tester l'égalité de **e2** et **e3** et afficher le résultat.
- 12- Créer une entreprise nommée **ent1** en fournissant le nom de l'entreprise.

- 13-** Ajouter huit employés à l'entreprise en utilisant les méthodes AddEmploye.
- 14-** Supprimer un employé de l'entreprise.
- 15-** Chercher un employé de l'entreprise.
- 16-** Calculer le salaire annuel.
- 17-** Augmenter le salaire des employés.
- 18-** Afficher l'entreprise.