

Laboratoire 4

La Programmation Orientée Objet avec C#

Exercice 1 : Rectangle

Partie 1 - Nous voulons représenter des formes géométriques particulière : Rectangle. Un rectangle est caractérisé par une largeur et une longueur de type réel. On vous demande :

- Créer la classe **Rectangle**
- Définir les attributs
- Définir les mutateurs et accesseurs pour chacun des attributs. Nous voulons que :
 - La longueur est un réel positif. Sinon on affecte 1.0 à la longueur comme valeur par défaut.
 - La largeur est un réel positif. Sinon on affecte 1.0 à la largeur comme valeur par défaut.
- Définir un constructeur initialisant la longueur.
- Définir un constructeur initialisant la longueur et la largeur.
- Définir une méthode nommée **Surface** permettant de calculer la surface du rectangle. La surface est définie par la formule suivante : largeur * longueur.
- Définir une méthode nommée **Perimetre** permettant de calculer le périmètre du rectangle. La surface est définie par la formule suivante : 2 * (largeur + longueur).
- Définir une méthode nommée **Afficher** permettant d'afficher le rectangle de la façon suivant :

Longueur : 5.0
Largeur : 2.0
Périmètre : 14.0
Surface : 10.0

- Définir une méthode nommée **Egalite** permettant de tester l'égalité du rectangle courant avec le rectangle passé comme paramètre. Deux rectangles sont égaux s'ils ont la même longueur et la même largeur.

Partie 2 - Nous voulons tester la classe Rectangle. On vous demande de :

- Créer un rectangle **r1** dont la longueur est égale à 5.
- Créer un rectangle **r2** dont la longueur est égale à 6 et la largeur est égale 3.
- Calculer et afficher la surface de **r1**.
- Calculer et afficher le périmètre de **r2**.
- Afficher **r1** et **r2**.
- Tester l'égalité de **r1** et **r2**.

Exercice 2 : Lampe

Partie 1 - Nous voulons représenter les lampes. Une lampe est caractérisée par une variable booléenne indiquant son état (allumée = true et éteint = false) et sa couleur. On vous demande :

- Créer la classe **Lampe**
- Définir les attributs
- Définir les mutateurs et accesseurs pour chacun des attributs.
- Définir un constructeur sans argument initialisant l'état de la lampe à éteint et la couleur à « jaune ».
- Définir un constructeur initialisant l'état. La couleur par défaut est « jaune ».
- Définir un constructeur initialisant tous les attributs.
- Définir une méthode nommée *Eteindre* permettant d'éteindre la lampe.
- Définir une méthode nommée *Allumer* permettant d'allumer la lampe.
- Définir une méthode nommée *Etat* permettant de retourner l'état de la lampe
- Définir une méthode nommée *Afficher* permettant d'afficher la lampe de la façon suivante :

Etat : éteint
Couleur : Rouge

Partie 2 - Nous voulons tester la classe Lampe. On vous demande de :

- Créer trois lampes **I1**, **I2** et **I3** avec les trois constructeurs.
- Éteindre **I1**.
- Allumer **I3**.
- Afficher l'état de **I2**.
- Afficher les lampes **I1**, **I2** et **I3**.

Exercice 3 : Employe & TestEmploye

Partie 1 - Nous voulons représenter les employés d'une entreprise. Un employé est caractérisé par son **nom**, son **prénom**, son **adresse**, son **année embauche** et son **salaire**. On vous demande de :

- Créer une classe **Employe**
- Ajouter les attributs
- Définir les mutateurs et accesseurs pour chacun des attributs. Nous voulons que :
 - Le nom soit en majuscule,
 - Le prénom soit en minuscule,
 - L'année d'embauche soit comprise entre 1995 et 2021. Sinon on affecte la valeur 1995 comme valeur par défaut.
 - Le salaire soit supérieur à 2000\$CAD. Sinon on affecte la valeur 2000 comme valeur par défaut.
- Définir un constructeur initialisant le nom, le prénom, le salaire et la date d'embauche.
- Définir un constructeur initialisant tous les attributs.

- Créer une méthode nommée `SalaireAnnuel` permettant de calculer le salaire annuel d'un employé.
- Créer une méthode nommée `Ancienneté` permettant de calculer l'ancienneté de l'employé. L'ancienneté est calculée par la formule suivante : $2021 - \text{Année embauche}$.
- Créer une méthode nommée `Prime` permettant de calculer la prime accordée à l'employé. La prime est calculée de la façon suivante :
 - Si l'ancienneté < 5 ans alors la prime est égale à 10% du salaire annuel.
 - Si $5 \leq \text{l'ancienneté} < 10$ ans alors la prime est égale à 15% du salaire annuel.
 - Si $10 \leq \text{l'ancienneté} < 15$ ans alors la prime est égale à 20% du salaire annuel.
 - Si l'ancienneté ≥ 15 ans alors la prime est égale à 25% du salaire annuel.
- Créer une méthode nommée `Afficher` permettant d'afficher un employé. L'affichage doit être comme suit :

Nom : Bouhlel
 Prénom : Mohamed Salah
 Salaire : 4500\$CAD
 Adresse : 123 Rue Charlemagne, Gatineau, A1B2C3, Canada.
 Tu as 16 années d'ancienneté

- Créer une méthode nommée `Equals` permettant de comparer L'employé courant avec l'employé passant comme paramètre.

Partie 2 - Créer une classe nommée ***TestEmploye*** permettant de tester les méthodes de la classe ***Employe***. On vous demande de :

- Créer l'employé (**emp1**) mohamed salah bouhlel avec un salaire de 4500\$ CAD, embauché en 2005 et habitant 123 Rue Charlemagne, Gatineau, A1B2C3, Canada.
- Créer l'employé (**emp2**) Eric duval avec un salaire de 2500\$CAD et embauché en 2015.
- Calculer et afficher le salaire annuel de **emp1**.
- Calculer et afficher le salaire annuel de **emp2**.
- Calculer et afficher l'ancienneté de l'employé **emp1**.
- Calculer et afficher la prime de l'employé **emp2**.
- Tester l'égalité de l'employé **emp1** avec l'employé **emp2**.
- Afficher les deux employés.