

## TP N°2

### Exercice 1 :

Considérons la classe Article qui est caractérisée par:

<b>Article</b>
- nom : <b>string</b>
- reference : <b>string</b>
- prixUnitaire : <b>double</b>
+ Article () ;
+ Article ( <b>string</b> _nom, <b>string</b> _reference, <b>double</b> _prixUnitaire) ;
+ Article (Article _article) ;
+ setNom( <b>String</b> nouveau_nom) : void
+ setRef( <b>String</b> ref) : void
+ setPrix( <b>double</b> nouveau_prix) : void
+ getNom() : String
+ getRef() : String
+ getPrix() : double
+ affiche() : void

- Créer la classe Article.
- Définir les constructeurs par défaut, avec argument.
- Définir les accesseurs(getters), les mutateurs (setters) et la méthode affiche().
- Dans main:
  - o Créer des objets de la classe Article avec les constructeurs par défaut et avec arguments.
  - o Afficher les attributs de ces objets en utilisant la méthode affiche().

### Exercice 2 :

Créer une classe Personne avec trois propriétés (Nom, Prénom et Age).

- Ajouter un constructeur d'initialisation.
- Ajouter la méthode getInfo() qui permet d'afficher les informations d'une personne.
- Stocker les objets Personne (5 personnes) dans une liste et les afficher en utilisant la boucle for et foreach.
- Faire les modifications nécessaires à votre programme pour permettre l'utilisateur de spécifier le nombre de personne à stocker puis lui donner la possibilité de renseigner les informations (nom,prenom et age).

Le programme doit afficher les informations des personnes. Puis les informations de la personne la plus âgée et la personne moins âgée et finalement l'âge moyen de toutes les personnes.

### Exercice 3 :

Une Voiture est caractérisée par sa marque, couleur, Matricule et Nombre de chevaux

- Créer la classe voiture

- Encapsuler tous les attributs (les rendre private puis ajouter les accesseurs et modifieurs), l'attribut Matricule ne possède pas sa méthode modifieur en effet on ne peut pas modifier la matricule d'une voiture
- Ajouter à la classe le constructeur qui permet d'initialiser la marque, la couleur, la matricule et le nombre de chevaux
- Ajouter la méthode présentation qui permet d'afficher les informations d'une voiture
- Sur la classe program.cs créer un tableau qui peut contenir cinq voitures puis remplir le tableau
- Écrire le code qui permet d'afficher les informations de toutes les voitures du tableau
- Écrire une fonction qui permet d'augmenter le nombre des chevaux de 1 cheval toutes les voitures du tableau
- Afficher les informations des voitures après modification
- Écrire le code qui permet d'inviter l'utilisateur pour saisir la matricule puis cherche et affiche les informations de la voiture relative, s'il n'existe pas il affiche le message «matricule inexistante»

#### **Exercice 4 :**

Créez une classe de base Employé avec les propriétés communes : Nom et Numéro d'employé.

- Créez une classe dérivée EmployéATempsPlein qui hérite de l'Employé. Cette classe devrait avoir la propriété SalaireMensuel et un constructeur pour initialiser ces propriétés.
- Créez une classe dérivée Consultant qui hérite également de l'Employé. Cette classe devrait avoir les propriétés: TauxHoraire et HeuresTravaillees et un constructeur pour initialiser ces propriétés.
- Créez un programme de test qui instancie des employés à temps plein et des consultants, et qui affiche les détails de chaque employé.