

Matière : Ingénierie des bases de données

Niveau : LSI-GL2

TP3 – Structures de données pour les bases de données

La manipulation des enregistrements d'une table de base de données par programmation se fait à travers les tableaux et les listes d'objets. L'objectif de ce TP est de maîtriser de telles structures de données. Les objets qui seront considérés représentent des articles.

Partie I : Classe Article

- 1- Donner la définition de la classe *Article* qui contient les trois attributs suivants : Reference (Entier), Designation (chaîne), Prix (réel). En plus des trois attributs, cette dernière dispose de deux constructeurs, un par défaut et l'autre paramétré. Elle dispose également de deux méthodes qui permettent respectivement de saisir et afficher les données relatives à un article.

```
class Article
{
    int Reference;
    string Designation;
    float Prix ;
    ... .. Article(){...}
    ... .. Article(...) { ... }
    ... .. Saisir() { ... }
    ... .. Afficher() { ... }
}
```

Partie II : La classe TabArticles

- 1- Compléter la définition de la classe *TabArticles* qui permet de gérer un ensemble d'articles à l'aide d'un tableau. La spécification partielle de cette classe est comme suit :

```
class TabArticles
{
    int NbArticles;
    Article[] LesArticles; // Tableau d'articles
    public TabArticles(int N) { ... }
    ... .. Saisir() { ... }
    ... .. Afficher() { ... }
}
```

- 2- Donner une fonction principale qui montre l'utilisation de cette classe.

Partie II : La classe ListArticles

Dans cette partie, la gestion d'un ensemble d'articles ne se fera pas à l'aide d'un tableau mais plutôt à l'aide d'une liste. Pour ce faire il est demandé de compléter la définition de la classe *ListArticles* suivante :

```
class ListArticles
{
    List<Article> LesArticles ; // Liste d'articles
    public ListArticles() { ... }      ... ..
    Ajouter(.....) { ... }
    ... .. Supprimer(.....) { ... }
    ... .. Afficher() { ... }
}
```

- La méthode *Ajouter* permet de saisir les données d'un article et de l'ajouter à la liste.
- La méthode *Supprimer* permet de saisir l'indice de l'article à supprimer et le supprime de la liste
- La méthode *Afficher* permet d'afficher la liste des articles.

Annexe : Les listes

- **List<T>** en C#.Net est une collection d'objets fortement typés qui peuvent être accédés par un index et possédant des méthodes pour faire le tri, la recherche et la modification.
- Elle est définie dans l'espace de noms : System.Collection.Generic

Création d'une Liste

List<T> permet de créer une liste d'objets de type T.

Syntaxe :

List<T> MaListe = new **List<T>**();

MaListe est une liste d'objets de type T.

Propriétés

Count : une propriété qui donne le nombre d'éléments de la liste.

[] : Accès à un élément de la liste. MaListe[i] **Principales**

méthodes :

Add() : permet d'ajouter un objet à la fin de la liste

RemoveAt(int ind) : permet de supprimer l'objet d'index ind.