

Matière: Ingénierie des bases de données

Filière: LSI-GL2

TP2 : Bases du langage C#

Exercice 1

Écrire un programme qui lit deux entiers, calcule leur somme et affiche le résultat à l'écran.

Exercice 2

Écrire un programme qui fait la saisie d'un tableau d'entiers, calcule la somme des éléments du tableau et affiche le résultat à l'écran.

Exercice 3

- 1- Reprendre le travail demandé dans les exercices 1 et 2 en utilisant la notion de fonction. On proposera ainsi les fonctions réalisant les tâches suivantes : Somme de deux entiers, Saisie d'un tableau, Somme des éléments d'un tableau. Ces fonctions seront implémentées sous la forme de méthodes associées à la classe contenant la fonction principale du programme.
- 2- Écrire une définition de la fonction principale (Main) qui illustre l'utilisation de ces méthodes.

Exercice 4

Soit la chaîne `str=" C# programming 2024 or Csharp programming 2024 "`.

Donner un programme qui :

- Affiche la taille de la chaîne `str`.
- Affiche la position de la première et la dernière occurrence du caractère 'C' dans la chaîne.
- Supprime les espaces de début et de fin.
- Copie la chaîne "C#" à partir de la chaîne `str`.
- Compte le nombre des chiffres dans la chaîne `str`.

Exercice 5

On veut gérer un groupe de personnes. Chaque personne est donnée par son nom. Le groupe sera créé sous la forme d'une liste (voir annexe). Donner le programme qui permet d'implémenter une telle liste. Le programme doit également inclure les opérations suivantes :

- Ajouter une nouvelle personne au groupe.
- Supprimer une personne par son nom.
- Rechercher une personne par son nom.
- Afficher toutes les personnes.
- Trier les personnes par ordre alphabétique.

Annexe : Les listes

En C#, une liste est une collection dynamique d'éléments fortement typés, définie dans l'espace de noms *System.Collections.Generic*. Contrairement aux tableaux, les listes peuvent changer de taille pendant l'exécution du programme, ce qui signifie qu'il est possible d'ajouter ou supprimer des éléments à tout moment.

Principales caractéristiques des listes en C# :

Dynamique : La taille d'une liste peut augmenter ou diminuer.

Indexée : Chaque élément de la liste est accessible via un index.

Générique : Vous pouvez créer des listes de n'importe quel type.

Création d'une Liste

List<T> permet de créer une liste d'objets de type T. T est un type générique qui sera instancié au moment de la création de la liste par un type concret. Par exemple List<int>, List<string>, etc.

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 :

- RemoveAt(int ind) : permet de supprimer l'objet d'index ind.
- Add(T item) : Ajoute un élément à la fin de la liste.
- Remove(T item) : Supprime la première occurrence d'un élément spécifique de la liste.
- Insert(int index, T item) : Insère un élément à une position spécifique dans la liste.
- Clear() : Supprime tous les éléments de la liste.
- Contains(T item) : Vérifie si un élément spécifique est présent dans la liste.
- IndexOf(T item) : Retourne l'index de la première occurrence d'un élément spécifique.
- Sort() : Trie les éléments de la liste.
- Reverse() : Inverse l'ordre des éléments dans la liste.
- AddRange(IEnumerable<T> collection) : Ajoute les éléments d'une collection à la fin de la liste.
- RemoveAt(int index) : Supprime l'élément à l'index spécifié.