



TP1 : Les bases de C#

Exercice 1

Ecrire un code C# qui retourne le signe du résultat de la multiplication de deux nombres entrés par l'utilisateur via la console sans calculer leur produit.

Les deux nombres sont des entiers différents de zéro.

Exercice 2

Ecrire un code C# qui demande à l'utilisateur de saisir une chaîne de caractères (un mois).

Affiche le nombre de jours de ce mois (28, 30 ou 31).

Exercice 3

Écrivez une fonction en C# nommée **CalculerSomme** qui prend un tableau d'entiers en entrée et renvoie la somme de tous les éléments du tableau.

Écrivez une deuxième fonction en C# nommée **SommeElementsInferieurs** qui prend un tableau d'entiers et une valeur seuil en entrée, puis renvoie la somme des éléments du tableau qui sont strictement inférieurs à la valeur seuil.

Utilisez ces fonctions pour calculer la somme totale des éléments d'un tableau donné, ainsi que la somme des éléments inférieurs à une valeur seuil.

Affichez le tableau ainsi que les résultats à l'utilisateur.

Exercice 4

Écrivez un programme en C# qui crée une matrice 3x3 d'entiers. Remplissez la matrice avec des valeurs de votre choix.

Affichez la matrice.

Écrivez une fonction nommée **CalculerMinimum** qui prend la matrice en entrée et renvoie la valeur minimale dans la matrice.

Écrivez une fonction nommée **CalculerMaximum** qui prend la matrice en entrée et renvoie la valeur maximale dans la matrice.

Écrivez une fonction nommée **CalculerSomme** qui prend la matrice en entrée et renvoie la somme de tous les éléments de la matrice.

Écrivez une fonction nommée **CalculerMoyenne** qui prend la matrice en entrée et renvoie la moyenne des éléments de la matrice.

Utilisez ces fonctions pour afficher le minimum, le maximum, la somme et la moyenne des éléments de la matrice.

Devoir :

Application de gestion de tâches (To-Do List)

1. Créez une application console C# qui permet à l'utilisateur de gérer une liste de tâches.
2. Les tâches doivent être stockées dans une liste.
3. L'application doit afficher un menu interactif avec les options suivantes :
 - Ajouter une nouvelle tâche
 - Afficher la liste des tâches
 - Marquer une tâche comme terminée

- Supprimer une tâche de la liste
 - Quitter l'application
4. Lorsque l'utilisateur ajoute une tâche, l'application doit lui permettre de saisir une description pour la tâche et l'ajouter à la liste des tâches.
 5. Lorsque l'utilisateur affiche la liste des tâches, l'application doit afficher toutes les tâches avec un numéro d'index pour chaque tâche (pour permettre la sélection ultérieure).
 6. L'utilisateur doit pouvoir marquer une tâche comme terminée en entrant le numéro de la tâche.
 7. L'utilisateur doit pouvoir supprimer une tâche en entrant le numéro de la tâche.
 8. L'application doit permettre à l'utilisateur de quitter à tout moment.
 9. Utilisez des fonctions pour organiser le code, de sorte que chaque fonction effectue une tâche spécifique (ajout, affichage, marquage, suppression) pour maintenir la lisibilité et la modularité du code.