



Cours : Programmation orientée objet - INF111

Enseignant : El Hachemi Alikacem

Exercices : Utilisation des structures de contrôle -Série I

Pour cette série d'exercice, vous pouvez créer un nouveau projet (nom de projet ca.inf111.(vos initiales).semaine2, par exemple) puis une classe Java par exercice.

1. Conditionnelles imbriquées

Exercice 1 :

Écrire un programme qui saisit deux nombres entiers représentants le numérateur et le dénominateur d'une fraction. Si le dénominateur est égal à 0 on affiche « impossible ». Sinon si le dénominateur est égal à 1 on affiche seulement le numérateur. Sinon si le numérateur est égal à 0 on affiche 0. Sinon si le dénominateur est négatif, on multiplie les deux entiers par -1 avant d'afficher numérateur/dénominateur

2. Traiter une série de nombres

Nous allons faire une série d'exercices qui traitent une série de nombres données par l'utilisateur. Ainsi, l'utilisateur va introduire les nombres un à un, quand il aura terminé, il introduit le caractère #.

Le code suivant permet de faire la lecture de la série de nombres. À utiliser pour les exercices suivants :

```
public static void main(String[] args) {
    String unInput ; // Donnée introduite par l'utilisateur
    Scanner clavier ;
    float unNombre ; // la valeur numérique de l'input

    clavier = new Scanner(System.in);

    System.out.print("Introduire une valeur numérique (ou # pour
terminer) : ");

    // Lecture du nombre sous la forme d'une chaîne de caractères
    unInput = clavier.nextLine() ;

    // On boucle tant que ce n'est pas "#"
    while (!unInput.equals("#")) {

        // Transformation de la chaîne de caractère en un nombre
        // Attention, il y aura une erreur si la chaîne de
        // caractères n'est pas un nombre
        unNombre = Float.parseFloat(unInput);

        // Affichage de la valeur numérique (non indispensable)
        System.out.println("La nouvelle valeur : " +unNombre);

        // -----
        // Mettre votre traitement ici ...
        // ...
        // -----

        // Lecture d'un nouveau nombre
        System.out.print("Introduire une valeur numérique (ou #
pour terminer) : ");
        unInput = clavier.nextLine();

    } // fin du while

    System.out.println("Au revoir !");
    clavier.close();
}
```

Exercice 2 :

Écrire un programme qui affiche le nombre des valeurs (données par l'utilisateur)

Exercice 3 :

Écrire un programme qui affiche la somme des valeurs.

Exercice 4 :

Écrire un programme qui affiche la somme des valeurs au carré

Exercice 5 :

Écrire un programme qui affiche la somme des valeurs paires

Exercice 6 :

Écrire un programme qui affiche le produits des valeurs non nulles

3. Traiter une série de nombres – Suite**Exercice 7 :**

En considérant une liste de valeurs données par l'utilisateur, écrire un programme qui détermine le nombre le plus grand de cette liste.

Utiliser le code de la section précédente.

Question : Qu'est-ce qu'il faut changer dans le programme, si on souhaite avoir le nombre le plus petit de la liste ?

4. Les séries mathématiques**Exercice 8 :**

Écrire un programme qui calcule la somme suivante :

$$S = 1 + 1/2 + 1/3 + 1/4 + \dots + 1/n$$

Exercice 9 :

Écrire un programme qui calcule la somme suivante :

$$S = 1/2 + 2/3 + 3/4 + \dots + n/(n+1)$$

Exercice 10 :

Écrire un programme qui calcule le produit somme suivant :

$$P = 1 * 2 * 3 * \dots * n$$

n : Valeur donnée par l'utilisateur