

TD 1

M.Adam – N.Delomez – JF.Kamp – L.Naert

3 août 2022

Objectifs du TD

- Structure simple d'un programme sous Java
- Notions d'erreurs de syntaxe
- notions d'erreurs à l'exécution
- Notions de variables, d'affectation, de séquentialité, d'alternative

Exercice 1 (*)

Soit le fichier Son.java :

```
/**
 * Vérifie la pression et le volume
 * @author M.Adam
 */
class Son {
    void principal() {
        final float pSeuil = 2.3;
        final float V_SEUIL = 7.41F

        System.out.println("Seuil pression : " + pSeuil + "\tSeuil Volume :  + V_SEUIL);

        double pression = SimpleInput.getFloat("Pression courante : ");
        float  Volume   = SimpleInput.getDouble("Volume courant : ");

        if (pression > pSeuil && Volume > V_SEUIL) {
            System.out.println ("pression ET Volume eleves. Stoppez !");
        } else if (presion > pSeuil)
            System.out.println ('Il faut augmenter le Volume');
        } else if (Volume > V_SEUIL) {
            System.out.println ("Vous pouvez diminuer le Volume");
        } else {
            System.out.println ("Tout va bien !");
        }
    }
}
```

1. Déterminer et corriger **toutes** les erreurs de syntaxe de ce programme.

Noter le nombre d'erreurs de syntaxe.

- 2.** Déterminer et corriger **toutes** les erreurs de convention de ce programme.

Noter le nombre d'erreurs de convention.

- 3.** Déterminer les valeurs de la pression et du volume pour tester le programme `Son.java`.

Exercice 2 (**)

Soit le programme `Racine.java` :

```
/**
 * Rôle à deviner
 * @author M.Adam
 */
class Racine{

    void principal() {
        double a, b, c, delta, racine1, racine2 ;

        a = SimpleInput.getDouble ("Valeur de a : ");
        b = SimpleInput.getDouble ("Valeur de b : ");
        c = SimpleInput.getDouble ("Valeur de c : ");
        delta =  b * b - 4 * a * c;
        racine1 = ((-1) * b - Math.sqrt (delta)) / (2 * a);
        racine2 = ((-1) * b + Math.sqrt (delta)) / (2 * a);
        System.out.println ("Racine 1 = " + racine1);
        System.out.println ("Racine 2 = " + racine2);
    }
}
```

La définition de `Math.sqrt()` est donnée par :

```
double sqrt(double a)
    Returns the correctly rounded positive square root of a double value.
```

1. Que calcule ce programme ?

2. Donner des valeurs de `a`, `b` et `c` qui donnent une valeur à `racine1` et `racine2`.

3. Que se passe-t-il si la valeur de a est zéro ?

4. Donner des valeurs non nulles de a, b et c qui provoquent une erreur à l'exécution.

5. Modifier le programme pour qu'il affiche le résultat uniquement si les valeurs de a, b et c permettent le calcul.

```
void principal() {  
    double a, b, c, delta, racine1, racine2 ;  
    a = SimpleInput.getDouble ("Valeur de a : ");  
    b = SimpleInput.getDouble ("Valeur de b : ");  
    c = SimpleInput.getDouble ("Valeur de c : ");  
    delta = b * b - 4 * a * c;
```


Exercice 3 (***)

Écrire un programme qui fait entrer à l'utilisateur un temps en secondes, et qui le convertit en une durée exprimée en heures, minutes et secondes. Vérifiez que la solution fonctionne entre autres avec 119s, soit 1 minute et 59 secondes.

- L'opérateur modulo en Java est `%`. Par exemple, `4 % 2` vaut 0 et `5 % 2` vaut 1.
- En java, la division de deux entiers donne un entier. Par exemple, `8 / 3` vaut 2 et `6 / 2` vaut également 3.

Exercice 4 (**)

Que calcule le programme suivant ?

```
/**
 * Rôle à deviner
 * @author M.Adam
 */
class ADeviner {
    void principal() {
        int a;
        a = SimpleInput.getInt ("Valeur de a : ");
        if (a % 4 == 0 && a % 100 != 0 ) {
            System.out.println( a + " respecte la propriété !");
        } else {
            if (a % 400 == 0) {
                System.out.println (a + " respecte la propriété !");
            } else {
                System.out.println (a + " ne respecte pas la propriété !");
            }
        }
    }
}
```


Exercice 5 (***)

Écrire le programme de l'exercice précédent d'une manière différente.

```
void principal() {  
    int a;  
    a = SimpleInput.getInt ("Valeur de a : ");
```