

MOOC Init Prog Java

Tutoriels semaine 1

Les tutoriels sont des exercices qui reprennent des exemples du cours et dont le corrigé est donné progressivement au fur et à mesure de la donnée de l'exercice lui-même.

Ils sont conseillés comme un premier exercice sur un sujet que l'étudiant ne pense pas encore assez maîtriser pour aborder par lui-même un exercice «classique».

Semaine 1 : Calcul de l'IMC

Nous voulons écrire un programme permettant de calculer l'IMC (indice de masse corporelle) d'une personne. L'IMC se calcule à partir du poids et de la taille suivant la formule : poids (en kg) divisé par le carré de sa taille (en m).

1. Commencez par ouvrir le fichier (vide) `IMC.java` dans votre éditeur favori ou dans Eclipse.
2. Préparez la «coquille vide» de base accueillant votre programme :

```
class IMC {  
    public static void main(String[] args) {  
    }  
}
```

Nous verrons plus tard dans le cours à quoi correspondent exactement toutes ces lignes, mais pour l'instant considérez cela comme la base minimale pour que votre programme fonctionne.

On peut maintenant commencer à attaquer notre problème.

3. Commençons notre programme en prévoyant des variables pour représenter les données de notre problème, à savoir un poids et une taille.

A priori ces grandeurs sont des nombres réels ; nous **déclarons** donc **des variables** de type `double` :

```
class IMC {  
    public static void main(String[] args) {  
        double poids;  
        double taille;  
    }  
}
```

4. Maintenant, le réflexe du bon programmeur : il faut penser à initialiser ces variables. Choisissons ici des valeurs raisonnables quelconques, par exemple :

```
class IMC {  
    public static void main(String[] args) {  
        double poids = 74.5 ;  
        double taille = 1.75 ;  
    }  
}
```



```

    }
}

```

8. puis en posant les questions :

```

import java.util.Scanner;
class IMC {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        double poids = 74.5 ;
        double taille = 1.75 ;

        System.out.print("Entrez un poids (en kg) : ");
        poids = scanner.nextDouble();
        System.out.print("Entrez une taille (en m) : ");
        taille = scanner.nextDouble();

        double imc = poids / (taille * taille);

        System.out.println("Pour " + poids
                           + " kg et " + taille
                           + " m, l'IMC est de " + imc);
    }
}

```

Notre programme peut maintenant calculer l'IMC pour n'importe quels poids et taille. Il peut être amélioré en contrôlant que les valeurs saisies soient bien raisonnables. Mais pour cela, il faut utiliser ce que l'on appelle des structures de contrôle (vues en semaine 2 du cours).

Concernant la lecture de données introduites au clavier, il convient de faire les remarques suivantes:

1. La solution montrée ci-dessus utilise le bagage actuel : celui avec lequel vous ne savez mettre des instructions que dans le corps de la méthode `main`. L'objet `scanner` de type `Scanner` qui permet de lire des données introduites depuis le terminal est déclaré et initialisé dans la méthode `main`. si vous procédez de cette manière, et que vous travaillez sous Eclipse, un message de mise en garde sera généré pour vous indiquer que `scanner` n'a pas été fermé. Les causes de ce message dépassent les notions vues dans le cadre de ce cours d'introduction et ce message n'aura aucune incidence sur l'exécution de votre programme. Pour l'éviter il suffira d'ajouter l'instruction `scanner.close()` ; avant la dernière accolade de `main`; c'est à dire juste après le dernier `System.out.println(...)` dans le code précédent. Une façon alternative de déclarer cette variable de type `Scanner` est présentée dans le corrigé du premier exercice.
2. Si l'introduction de valeur de type `double` avec un point décimal provoque une erreur d'exécution («`InputMismatchException`») c'est probablement que votre environnement est configuré avec des «Locales» (paramètres régionaux) non anglo-saxons. Vous pouvez faire en sorte qu'une séquence particulière d'entrées se fasse selon les conventions anglo-saxonnes. Ceci se fait en ajoutant l'instruction `nomDeLaVariableScanner.useLocale(Locale.ENGLISH)` avant l'utilisation de `nomDeLaVariableScanner` (donc dans notre exemple ci-dessus `nomDeLaVariableScanner` serait à remplacer par `scanner`). Il faut pour cela avoir écrit la directive d'importation suivante en début de programme : `import`

```
java.util.Locale;
```
