

IPO – TP 1 : syntaxe et rappels impératifs

<http://www.lri.fr/~blsk/IPO/>

Mise en route

1. Récupérer le fichier HelloWorld.java sur la page du cours,
2. compiler le fichier en utilisant : `javac HelloWorld.java` dans un terminal, en vous plaçant dans le répertoire où vous avez mis le fichier,
3. exécuter le programme avec : `java HelloWorld` dans un terminal,
4. ouvrir le fichier avec un éditeur pour modifier l’affichage (lui faire afficher votre nom par exemple), puis compiler et exécuter.

Vous trouverez également sur la page un squelette de programme utilisable pour les exercices suivants (pensez à changer le nom du fichier et le nom du programme), ainsi qu’un fichier vous donnant des exemples de saisies clavier.

5. Consulter le fichier ScannerDemo.java sur la page du cours,
6. modifier le programme HelloWorld pour qu’il demande son nom à l’utilisateur avant de le saluer.

Exercice 1 – Expressions

1. En utilisant la formule $C = \frac{5}{9}(F - 32)$, écrire un programme Degre qui lit une température exprimée en degrés Fahrenheit et affiche sa valeur en degrés Celsius.

Exemples d’exécution du programme

```
$java Degres
Donnez une température en Fahrenheit : 0.0
Cette température équivaut à -17.8 degrés Celsius
$java Degres
Donnez une température en Fahrenheit : 60.0
Cette température équivaut a 15.6 degrés Celsius
```

2. Écrire un programme Hjms qui, pour un nombre de secondes donné, calcule et affiche son équivalent en nombre de jours, d’heures, de minutes et de secondes.

Exemples d’exécution du programme

```
$java Hjms
Donnez une durée en secondes : 3621
Cette durée équivaut à 0 jour 1 heure 0 minute 21 secondes
$java Hjms
Donnez une durée en secondes : 180181
Cette durée équivaut à 2 jours 2 heures 3 minutes 1 seconde
```

Exercice 2 – Instructions conditionnelles

1. Écrire un programme Couronne qui, pour un point P du plan de coordonnées x et y demandées à l’utilisateur, détermine si ce point se trouve ou non à l’intérieur de la couronne de centre $(0, 0)$ et définie par un rayon extérieur r_{ext} et un rayon intérieur r_{int} .

Exemple d’exécution du programme

```
$java Couronne
Rayon extérieur : 14
Rayon intérieur : 10
Donnez un point x : 12 y : 0
Ce point est dans la couronne.
```

2. Écrire un programme `TroisNombres` qui lit trois nombres au clavier, les classe dans l'ordre croissant et les affiche du plus petit au plus grand.

Exemple d'exécution du programme

```
$java TroisNombres
1er nombre : 14
2e nombre : 10
3e nombre : 17
Les nombres dans l'ordre croissant : 10 14 17
```

Exercice 3 – Boucles

1. Écrire un programme `Triangle` qui affiche un motif triangulaire dont la taille est fixée par une valeur donnée au clavier.

Exemple d'exécution du programme

```
$java Triangle
Donnez la taille du motif : 7
*
**
***
****
*****
*****
*****
```

- (a) Écrire une première version en n'utilisant que des boucles **for**,
- (b) puis une deuxième version en n'utilisant que des boucles **while**.

2. Écrire un programme `Pyramide` qui affiche un motif pyramidal dont la taille est fixée par une valeur donnée au clavier. Utilisez les boucles qui semblent les mieux adaptées.

Exemple d'exécution du programme

```
$java Pyramide
Donnez la taille du motif : 7
      *
     ***
    *****
   *****
  *****
 *****
*****
```

Exercice 4 – Suite de Fibonacci La suite de Fibonacci est définie par la formule de récurrence suivante :

$$\begin{cases} u_0 = 0 \\ u_1 = 1 \\ u_n = u_{n-1} + u_{n-2} & \text{si } n \geq 2 \end{cases}$$

1. Écrire un programme `Fibo1` qui permet de calculer le n^{e} terme de la suite de Fibonacci, n étant fixé par l'utilisateur.
2. Écrire un programme `Fibo2` qui permet d'obtenir la valeur et le rang du premier terme de cette suite supérieur à une valeur limite donnée par l'utilisateur.