

**Exercices du module**  
**Renforcement Technique JAVA**

## → Important !

Vous devrez rendre vos exercices sur un dépôt GitHub que vous aurez préalablement créé. Il devra être public et s'appeler « **JavaExercises\_[votre nom-votre prénom]** ». Par exemple : « **JavaExercises\_dupont-martin** ».

Une fois votre dépôt cloné sur votre ordinateur, vous devrez ajouter un dossier nommé « **file1** » à l'intérieur. Chacun de vos exercices devra être dans un fichier distinct et déposé dans ce dossier. Vous nommerez les fichiers « **exercice1.java** », « **exercice2.java** », etc.

### Rappel des commandes Git essentielles à utiliser sur votre terminal :

- git clone [nom de dépôt]
- git add [nom de fichier]
- git commit -m "[description de vos modifications]"
- git push origin main
- git status
- git log

## ⇒ Préambule

Dans ces exercices vous allez apprendre à manipuler des variables, des tableaux, des conditions, des boucles et des affichages dans le terminal.

**Attention :** Les types de variables dans les exercices peuvent ne pas être précisé, dans ces cas-là, vous devrez définir le type le plus adapté à la situation !

## ⇒ Exercice 1

Écrivez une variable « **String** » appelée « **name** », laquelle a pour valeur « **Alex** ». Ajoutez une autre variable « **Integer** » qui s'appelle « **age** » et qui contient la valeur « **25** ». Affichez ensuite le contenu des deux variables avec la fonction « **println** ».

**Attention :** La première variable contient une chaîne de caractère et la seconde un nombre.

Résultat attendu :

```
java exercise1  
Alex25
```

## ⇒ Exercice 2

Écrivez une variable « **double** » appelée « **price** », laquelle a pour valeur « **19.99** ». Ajoutez une autre variable « **Integer** » qui s'appelle « **quantity** » et qui contient la valeur « **12** ».

Ensuite, créez une variable « **total** » qui ait pour valeur le résultat de la multiplication des valeurs de « **price** » et de « **quantity** ».

Enfin, affichez « **total** » grâce à la fonction « **println** ».

Résultat attendu :

```
java exercise2  
239.88
```

### ⇒ Exercice 3

Écrivez les variables suivantes :

- « **name** » a pour valeur « **Halil** »
- « **age** » contient la valeur « **27** »
- « **job** » possède la valeur « **développeur** »

Affichez le contenu des variables avec la fonction « **println** » afin de former les phrases suivantes : « Je m'appelle Halil, j'ai 27 ans et je suis développeur. »

**Attention :** Vous devez utiliser une seule fois l'instruction « **println** », pensez à vous servir de l'opérateur « **+** » pour cela.

Résultat attendu :

```
java exercise3
Je m'appelle Halil, j'ai 27 ans et je suis développeur.
```

### ⇒ Exercice 4

Écrivez une variable appelée « **str** » qui a pour valeur « **Hello World !** ». Il en faudra également une nommée « **i** » qui sera égale à **0**.

Vous devez utiliser une boucle **while()** qui va vérifier que « **i** » est strictement inférieur à la longueur de « **str** ». Vous afficherez les caractères de « **str** » un à un, séparés par un caractère espace, et incrémenterez « **i** » de 1 à chaque fois. Les espaces ne doivent être affichés que entre deux caractères, pour cela vous devrez utiliser une condition pour savoir si vous n'êtes pas sur le dernier caractère de la phrase.

**Attention :** La méthode « **length()** » peut vous être utile pour calculer la longueur de « **str** ».

Résultat attendu :

```
java exercise4
H e l l o   W o r l d   !
```

### ⇒ Exercice 5

Écrivez une variable appelée « **str** » qui a pour valeur « **abcdefghijklmnopqrstuvwxyz** ». Il en faudra également une nommée « **i** » qui sera égale à 0.

Vous devez utiliser une boucle **while()** qui va vérifier que « **i** » est strictement inférieur à la longueur de « **str** ». Vous afficherez un caractère de « **str** » sur deux et incrémenterez « **i** » de 1 à chaque fois.

**Attention :** L'utilisation de l'opérateur modulo peut s'avérer très efficace !

Résultat attendu :

```
java exercise5
bdfhjlnprtvxz
```

#### ⇒ Exercice 6

Écrivez une variable appelée « **str** » qui a pour valeur « **abcdefghijklmnopqrstuvwxyz** ». Il en faudra également une nommée « **i** » qui sera égale à la longueur de « **str** ».

Vous devez utiliser une boucle **while()** qui va vérifier que « **i** » est supérieur ou égal à 0. Vous décrémenterez « **i** » de 1 à chaque fois.

**Attention :** Faites attention à l'index car un tableau commence par « 0 ».

Résultat attendu :

```
java exercise6
zyxwvutsrqponmlkjihgfedcba
```

## ⇒ Exercice 7

Écrivez une boucle « `for(;;)` » qui va avoir en paramètre une variable « `i` » égale à 0 et dont la valeur incrémentera à chaque tour dans la limite de 10. Vous afficherez la table de multiplication de 5 en respectant le modèle d'affichage attendu.

Résultat attendu :

```
java exercise7
0 x 5 = 0
1 x 5 = 5
2 x 5 = 10
3 x 5 = 15
4 x 5 = 20
5 x 5 = 25
6 x 5 = 30
7 x 5 = 35
8 x 5 = 40
9 x 5 = 45
10 x 5 = 50
```

## ⇒ Exercice 8

Écrivez une variable « `colors` » qui est égale à un tableau à 3 valeurs : « bleu », « rouge », « jaune ». Les tableaux sont définis par le type suivis des symboles « `[]` ». Ensuite, utilisez une boucle « `foreach` » ( `for( : )` ) qui va parcourir le tableau et afficher chaque couleur dans une phrase en particulier sous le format d'une variable « `value` ». Il faudra afficher « `value` » à la suite des phrases en fonction de sa valeur.

**Attention :** A l'intérieur de la boucle, une comparaison est attendu pour afficher la bonne phrase par rapport à la couleur !

Résultat attendu :

```
java exercise8
La couleur préférée de Louis est le bleu
Pour sa peinture, Andrée a utilisé du rouge
Anis n'a pas de chemise jaune
```

#### ⇒ Exercice 9

Écrivez un tableau nommé « **numbers** » contenant les valeurs suivantes : 21565, 3412, 180, 1556, 182, 84, 15.

Ajoutez une boucle « **foreach** » ( **for( : )** ) qui va afficher uniquement les nombres qui ne sont pas divisibles par 3. Vous placerez des virgules entre les chiffres affichés de façon qu'il n'y en ait pas une avant le premier ni après le dernier.

**Attention :** Il y a deux façons de procéder pour l'affichage des virgules. Essayez de gérer le cas où la première valeur du tableau serait divisible par 3 si vous voulez aller plus loin !

Résultat attendu :

```
java exercise9
21565, 3412, 1556, 182
```