

# **Java : Partie 2**

## **Les Bases du langage**

### **Exercices**

Juin 2013

Auteur : Chouaïb LAGHLAM

Concepteur Développeur Informatique  
Module : Programmation Java  
Bases du Langage - Exercices

---

**Bases du Langage Java : Exercices**

**Préalable     3**

Exercice 00010	:	3
Exercice 00020	:	3
Exercice 00030	:	3
Exercice 00040	:	4
Exercice 00050	:	4
Exercice 00060	:	4
Exercice 00070	:	5
Exercice 00080	:	5
Exercice 00090	:	5
Exercice 00100	:	6
Exercice 00110	:	6
Exercice 00120	:	7
Exercice 00130	:	7
Exercice 00140	:	8
Exercice 00150	:	9
Exercice 00160	:	9

# Concepteur Développeur Informatique

## Module : Programmation Java

### Bases du Langage - Exercices

---

### Préalable

Dans votre l'espace de travail « **Java\_010\_EspaceTravailCours** », dans le projet « **prj\_Java\_010\_LesBases** », créez un package « **pack\_20\_LesBases\_Exos** »,

Pour chaque exercice, créez une classe java qui rappelle le nom de l'exercice et qui est dotée de la méthode main.

Par exemple :

Pour l'exercice 00010, créez la classe « Ex00010 » dotée de la méthode main

#### **Exercice 00010 :**

Ecrire un programme qui examine les 20 premiers entiers et décide pour chacun s'il est : pair ou multiple de 3 ou multiple de 5.

#### **Exercice 00020 :**

Ecrire un programme qui demande un nom et un login et décide si l'utilisateur peut ouvrir un session ou pas

Comparer avec deux valeurs constantes supposées être les bonnes valeurs.

#### **Exercice 00030 :**

Ecrire un programme qui calcule la racine carrée d'un nombre saisi.

Le programme fonctionne en boucle jusqu'à ce que l'utilisateur saisisse zéro (signe qu'il faut arrêter La boucle et le programme)

Le programme refusera les valeurs négatives.

Son exécution se présentera ainsi :

```
Donnez un nombre positif : 2
Sa racine carrée est      : 1 4142135
Donnez un nombre positif : -3
Svp positif
Donnez un nombre positif : 5
Sa racine carrée est      : 2 236067
Donnez un nombre positif : 0
```

Le programme s'arrête.

# Concepteur Développeur Informatique

## Module : Programmation Java

### Bases du Langage - Exercices

---

#### Exercice 00040 :

Ecrire un programme qui calcule la somme des  $n$  premiers termes de la «**série harmonique**» .  
C'est-à-dire la somme de  $1 + 1/2 + 1/3 + 1/4 + 1/n$   
La valeur  $n$  sera lue en donnée.

#### Exercice 00050 :

Ecrire un programme qui demande le nom d'une saison (été, printemps, ....) et qui affiche une phrase selon la saison  
(Par exemple pour le printemps : « les fleurs commencent à sortir,...»)

#### Exercice 00060 :

Ecrire un programme qui affiche un triangle isocèle formé d'étoiles.  
La hauteur du triangle (c'est à dire son nombre de lignes) sera fournie en saisie.  
Comme dans l'exemple ci-dessous, on s'arrangera pour que la dernière ligne s'affiche sur le bord gauche de l'écran :

```
combien de lignes ? 8
      *
     ***
    *****
   *********
  ***********
 *****
*****
*****
*****
```

# Concepteur Développeur Informatique

## Module : Programmation Java

### Bases du Langage - Exercices

---

#### Exercice 00070 :

Ecrire un programme qui affiche toutes les manières possibles d'obtenir un euro avec des pièces 2 centimes, 5 centimes et 10 centimes.

Dire combien de possibilités ont ainsi été trouvées.

Les résultats sont présentés

Ainsi :

$$1 \text{ €} = 50 \times 2 \text{ c}$$

$$1 \text{ €} = 45 \times 2 \text{ c} + 2 \times 5 \text{ c}$$

$$1 \text{ €} = 40 \times 2 \text{ c} + 4 \times 5 \text{ c}$$

En tout, il y aura... Façons de faire 1 €

**Remarque :** évidemment, c'est par programmation que vous trouverez les différentes combinaisons. Ne pas se contenter d'afficher du texte.

#### Exercice 00080 :

Ecrire un programme qui crée un tableau d'entiers et le remplit par les carrés des N premiers nombres impairs.

La valeur N étant lue au clavier.

Le programme, ensuite, affiche le contenu du tableau.

Exemple :

Combien de valeurs : 5

1 a pour carré 1

3 a pour carré 9

5 a pour carré 25

7 a pour carré 49

9 a pour carré 81

Remarque : L'utilisation d'un tableau est obligatoire dans cet exercice.

#### Exercice 00090 :

Une entreprise de location de véhicules de tourisme applique les tarifs suivants :

- La journée de location coûte 75 Euros HT à la base.
- Les 100 premiers kilomètres sont gratuits.
- Les 200 Km supplémentaires coûtent 0,5 Euros HT du kilomètre.
- Les kilomètres suivants sont facturés 0,3 Euros HT du kilomètre.

En plus de la méthode « main », créez les méthodes suivantes :

# Concepteur Développeur Informatique

## Module : Programmation Java

### Bases du Langage - Exercices

---

- La méthode «**calculTTC**» calculant le montant TTC en fonction du montant HT et de la TVA passés en paramètres.
- La méthode «**APayer**» (utilisant la méthode «calculTTC») pour calculer le tarif TTC à payer : Elle reçoit le nombre de jours de location et le nombre de km parcourus et renvoie le montant à payer

Codez, alors la méthode « main » pour permettre de saisir la durée et le kilométrage et d'afficher le tarif (la TVA est fixée d'office à 19,60%).

(la méthode « main » appelle évidemment la méthode « APayer »).

**Remarque** : pour vérifier l'exactitude de vos calculs, comparez vos résultats à ceux-ci :

Nbre jours location	Nbre km parcourus	Total HT	Total TTC à payer
10	750	985.0	1178.06
10	300	850.0	1016.6
1	80	75	89,7
1	120	85.0	101,66

#### Exercice 00100 :

Un professeur souhaite saisir les notes d'un contrôle de maths et à la fin de la saisie (entrée de zéro), voir s'afficher la moyenne.

Je vous propose de réaliser cette tâche de la façon suivante à respecter :

➔ La déclaration d'un tableau de décimaux et le remplissage de ce tableau par la saisie des notes sera faite Dans la méthode «main»,

➔ Le calcul de la moyenne est à faire dans une méthode «**calMoyenne**» :

- Cette méthode reçoit le tableau des notes,
- Cette méthode renvoie la moyenne calculée.

➔ La méthode « main » appelle la méthode « calMoyenne » et affiche la moyenne

#### Exercice 00110 :

Vous savez que le taux de TVA, le prix HT et TTC sont liés par la relation :  $TTC = HT * (1 + TVA / 100)$  si la TVA est exprimée en pourcentage.

Ainsi, s'il on connaît deux valeurs sur trois, on peut calculer celle qui manque.  
C'est ce que vous allez faire.

- ➔ Créez, en plus de la méthode «main» et la méthode «**calculManquant** » prenant trois paramètres : montant TTC, montant HT et taux TVA :
- Vous supposerez que deux paramètres sur les trois seront initialisés à des valeurs valides.

# Concepteur Développeur Informatique

## Module : Programmation Java

### Bases du Langage - Exercices

---

- La méthode doit calculer la valeur non initialisée en fonction des deux autres et modifier le paramètre en conséquence.
  - Comment connaître le paramètre non initialisé et donc à modifier ?  
Convenons de passer -1 dans le paramètre manquant  
(Aucune confusion car -1 n'est pas une somme ou un taux valide).
- ➔ Codez la méthode «**main**» pour qu'elle appelle la méthode «calculManquant » et teste le calcul de la valeur manquante

Exemple :

On appelle la méthode « calculManquant » en lui passant : 119,6                      100,00 et                      -1  
Cela signifie qu'elle reçoit un TTC, un HT et il faut qu'elle calcule le taux tva : elle doit donc renvoyer 19,6

### Exercice 00120 :

Une librairie qui fait du commerce en ligne souhaite se doter d'une application calculant le montant du port à payer pour les commandes qu'elle expédie.

Voici les règles de gestion:

- Si le nombre d'articles commandés est inférieur à 3, le port est de 2,5 Euros par ouvrage.
  - Si le nombre est compris entre 3 et 5, le port sera égal au plus petit des deux montants suivants :
    - i) 2.5 Euros par ouvrage.
    - ii) 10% du total de la commande.
  - Si le nombre est compris entre 6 et 10, le port est un forfait de 15 Euros.
  - Si le nombre est supérieur à 10, les frais de port sont gratuits.
- ➔ Créez, en plus de la méthode «**main**» :
- La méthode «**portAPayer**» prenant deux paramètres : montant commande, nombre ouvrages,
- ➔ Codez la méthode «main» pour qu'elle appelle la méthode «portAPayer » et affiche la valeur de ce port.

### Exercice 00130 :

Vous devez maintenant écrire un programme calculant le bénéfice que réalise une entreprise sur un produit donné.

On vous fournit le code du produit, le prix de vente et la quantité vendue.

Pour connaître la marge réalisée sur le prix de vente, il faut se référer au code produit qui est une chaîne de caractères:

- Si le code débute par B, la marge est de 7 %;
- Cas particulier, la marge est de 10 % pour les produits dont le code commence par les caractères BZ;
- Si le code commence par AE, AR ou AT, la marge est de 8 %;
- Cas particulier, la marge est de 5 % si le code débute par AEP;

# Concepteur Développeur Informatique

## Module : Programmation Java

### Bases du Langage - Exercices

---

- Dans tous les autres cas, la marge est de 6 %.

Prenons l'exemple du produit codé AT554. Comme son code commence par AT, la marge est de 8%.  
Si l'entreprise vend 200 exemplaires de ce produit à 65 Euros pièce, elle réalise un bénéfice de  
 $8\% * 200 = 1040$  Euros.

- ➔ Créez, en plus de la méthode «main» :
  - La méthode «**calculBenefice** » prenant trois paramètres : code Produit, prix de vente et quantité vendue,
- ➔ Codez la méthode «main» pour qu'elle appelle la méthode «calculBenefice » et affiche le bénéfice réalisé.

### Exercice 00140 :

Il s'agit d'écrire une méthode statique qui permet de trier un tableau d'entiers.

En plus de la méthode «main», créez la méthode «trier» qui reçoit un tableau d'entiers et le tri du plus petit nombre au plus grand et renvoie le tableau trié.

Le tri est à faire par un algorithme de tri de votre choix.

La méthode « main » fait ceci :

- ➔ Elle déclare le tableau d'entiers,
- ➔ Demande la saisie des entiers et les stocke dans le tableau,
- ➔ Affiche le contenu du tableau non trié,
- ➔ Appelle la méthode « trier » et réceptionne le tableau trié,
- ➔ Affiche le tableau trié.



# Concepteur Développeur Informatique

## Module : Programmation Java

### Bases du Langage - Exercices

---

#### Exercice 00150 :

Vous allez demander à l'utilisateur de saisir une phrase puis vous lui affichez un tableau récapitulatif comme suit :

```
=====
```

Caractère	*	code ASCII	*	nombre occurrences
=====				
A	*	65	*	3
=====				
B	*	66	*	0
=====				
...etc.				
=====				
1	*	49	*	4
=====				
....etc.				

#### Exercice 00160 :

On va reprendre le jeu «**trop grand, trop petit**»,

Votre programme génère aléatoirement un nombre entier compris entre 1 et 100 et le stocke en mémoire sans le montrer à l'utilisateur,

L'utilisateur saisit un nombre au hasard entre 1 et 100. Votre programme lui «trop grand» ou «trop petit» en comparant le nombre saisi au nombre généré.

Lorsque l'utilisateur finit par trouver, il faut lui afficher :

«**Bravo, vous avez trouvé le nombre caché en ... tentatives**»