Classes, instances, objets [oo]
Exercices de cours

Karine Zampieri, Stéphane Rivière

Unisciel algoprog Version 20 mai 2018

Table des matières

Java - Exercices de cours (TP)

Mots-Clés Classes, instances, objets ■
Difficulté • ○ ○ (1 h 30) ■

DEPLACEZ Le Tour de magie dans oo03 ■

1 Appréhender le cours

1.1 Classe Personne / pgpersonne

Écrivez une classe `Personne` munie d'un `nom` (chaîne de caractères) et d'un `age` (entier).

Écrivez un constructeur qui initialise les attributs.

Écrivez des accesseurs `getNom` du nom et `getAge` de l'âge.

Écrivez un mutateur du nom `setNom(nom)`.

Écrivez une méthode `feterAnniv` qui fait vieillir la personne d'une année.

Écrivez un programme qui instancie une `Personne` de nom `"toto"` qui a 10 ans.

Affichez son nom et son âge.

Faites vieillir la personne jusqu'à 21 ans (exclus) en affichant un point `(.)` à chacun de ses anniversaires puis à 21 ans, elle se marie.

Modifiez son nom en suffisant `-mariage` (à son nom), puis affichez le nouveau nom.

Faites vieillir la personne de 5 ans en affichant un point `(.)` à chacun de ses anniversaires.

Finalement affichez son nom et son âge.

Testez. Résultat d'exécution :

```
toto a 10 ans
quelques annees plus tard
...........
A 21 ans il se marie
Nouveau nom: toto-mariage
cinq annees plus tard
.....
toto-mariage a maintenant 26 ans
```

1.2 Stockage en mémoire / qstkage

Objectif

Le but de cet exercice est de comprendre les différentes façons de stocker en mémoire une variable de type primitif et une variable de type objet.

Programme Java

On considère le programme suivant :

```
public class QZStkageA1{
  public static void main(String[] args){
    System.out.println("en A: -----");
    int[] k = new int[1];
    k[0] = 1;
    afficher(k[0]);
    modifier(k);
    afficher(k[0]);
  }

  static void modifier(int[] k){
    ++k[0];
  }

  static void afficher(int k){
    System.out.println(k);
  }
}
```

Expliquez le rôle des deux procédures.

Écrivez une classe `A` contenant un entier `k`.

Écrivez le constructeur.

Écrivez la méthode `modifier` équivalente à la procédure `modifier(k)`.

Écrivez la méthode `afficher` équivalente à la procédure `afficher(k)`.

Réécrivez le programme en utilisant la classe `A` de sorte à obtenir le résultat d'exécution suivant :

```
en B: -----
1
2
```

Java

On considère maintenant les instructions suivantes :

```
Stkage a2 = new Stkage(2);
Stkage a3 = new Stkage(3);
```

Qu'affiche le code suivant ?

```
System.out.println("en C: -----");
a2.afficher();
a3.afficher();
```

De même, qu'affichera :

```
System.out.println("en D: -----");
a3 = new Stkage(a2);
a2.afficher();
a3.afficher();
```

Enfin qu'affiche ce code ?

```
System.out.println("en E: -----");
a3.modifier();
a2.afficher();
a3.afficher();
```

1.3 Le grille pain erroné / qzxconstr1

Objectif
Il s'agit de corriger un programme.

Téléchargez le programme suivant :

C++ @[qzxconstr1.cpp]

Celui-ci contient la définition d'une classe `GrillePain` qui sert à représenter des grille-pains sous forme d'objets dotés de deux variables d'instances : l'`annee` de fabrication et le nombre de tranches `nbt` que l'appareil peut griller. Il y a également une méthode `afficher` que l'on peut appliquer sur un objet de type `GrillePain` afin d'afficher les valeurs de ses attributs.

Compilez le programme. Vous aurez des messages d'erreurs.
Pourquoi ? Corrigez et recompilez.

L'affichage lors de l'exécution (voir ci-dessous) montre que la variable d'instance `nbt` pour les deux grilles-pains ne possède pas la valeur envoyée au constructeur ? Pourquoi ? Corrigez le programme.

```
> GrillePain
GrillePain : annee = 1995 et nbt = 1
> GrillePain
GrillePain : annee = 1998 et nbt = 1
< GrillePain
< GrillePain
```

Qu'affiche la version corrigée du programme ?

2 Appliquer le cours

2.1 Apéritif / pgaperitif

Objectif
Cet exercice est relatif aux constructeurs et destructeur.
Soit le programme suivant :

Écrivez une classe `Aperitif` de sorte qu'il affiche le texte suivant :

```
L'heure de l'apero a sonne !
Super !
Encore un ?
Non merci.
A table !
```

2.2 Un peu de douceur / pgfleur

Objectif
Cet exercice est relatif aux constructeurs et destructeur.
Soit le programme suivant :

Écrivez une classe `Fleur` de sorte qu'il affiche le texte suivant (d'après un poème arabe du 5è siècle de l'HÉGIRE) :

```
Violette fraichement cueillie
Fragile corolle taillee
dans un cristal veine de bleu
Donne un poeme un peu fleur bleue
ne laissant plus qu'un simple soufle...
qu'un simple soufle...
```

La solution n'est pas forcément unique.

3 Approfondir le cours

3.1 Classe Consom / pgconsom

Objectif
Cet exercice traite des structeurs (constructeur, constructeur de recopie, destructeur). Il est relatif à la consommation en combustible d'un véhicule. Celle-ci est caractérisée par :

- Le nombre de litres/100 km (réel) consommé en trajet urbain.
- Le nombre de litres/100 km (réel) consommé en trajet autoroutier.
- La capacité (réel) de son réservoir (« autonomie »).

Écrivez une classe `Consom` représentant des consommations en carburant. Munissez-la des attributs spécifiés.

Rappel de cours
Les attributs de votre classe doivent être privés.

Écrivez un constructeur qui **initialise** les attributs et **affiche** le message « Vroum ».

Écrivez un constructeur de recopie qui affiche le message « re-Vroum ».

Écrivez le destructeur qui affiche le message « Arret moteur ».

Écrivez une méthode `eval(px,vkm,akm)` qui, pour le prix (par litre) `px` (réel) d'un combustible, un nombre de kilomètres effectuées en ville `vkm` (réel) et un nombre de kilomètres effectués sur autoroute `akm` (réel), calcule et renvoie l'évaluation du prix du trajet. La formule est la suivante :

$$(vkm * consomUrbaine/100 + akm * consomAutoroute/100) * px$$

Écrivez une méthode `afficher` qui affiche les caractéristiques de la consommation (où `[x]` désigne le contenu de `x`) :

```
consommation ville: [consomUrbaine] litres/100km
consommation autoroute: [consomAutoroute] litres/100km
autonomie: [autonomie] litres
```

Écrivez un programme qui définit la constante `PRIX_ESSENCE` (prix de l'essence) de valeur 1.5.

Instanciez une voiture `v1` de paramètres de consommation `(10.5,7.5,50.0)`.

Instanciez une voiture `v2` par recopie de `v1`.

Évaluez et affichez le prix du trajet de `v1` pour 20 km en ville et 100 km sur autoroute.

Affichez les caractéristiques de `v2`.

Testez. Résultat d'exécution :

```
Vroum
re-Vroum
Prix trajet v1 (en euros):
  20km/ville et 100km/autoroute: 14.399999999999999
Voiture v2:
  consommation ville: 10.5 litres/100km
  consommation autoroute: 7.5 litres/100km
  autonomie: 50.0 litres
Arret moteur
Arret moteur
```

3.2 Petit tour de magie / pgmagicien

Objectif

Cet exercice simule un tour de magie élémentaire suivant :

> Un magicien demande à un spectateur d'écrire sur un papier son âge et la somme qu'il a en poche (moins de 100 €). Il demande ensuite de montrer le papier à son assistant, qui doit le lire (sans rien dire), puis effectuer secrètement le calcul suivant : multiplier l'âge par 2, ajouter 5, multiplier le résultat par 50, ajouter la somme en poche, et soustraire le nombre de jours que contient une année, puis finalement donner le résultat à haute voix. En ajoutant mentalement 115 au chiffre reçu, le magicien trouve l'âge et la somme en poche (qui étaient restés secrets).

Modélisez ce tour de magie, en définissant les classes (simples) `Magicien`, `Assistant` et `Spectateur`. Il pourrait également être utile de disposer d'une classe `Papier`. Pour chaque méthode, effectuez un affichage à l'écran de l'opération en cours et de l'acteur qui la réalise.

Aide simple

Il existe de nombreuses variantes. Commencez par un modèle très simple et faites le évoluer pour vous rapprochez de la situation décrite.

Écrivez un programme qui simule le jeu. L'instance de `Spectateur` devra demander son âge à l'utilisateur ainsi que la somme d'argent en poche et s'assurer de la validité de la valeur (entre 1 et 99).

Testez. Exemple d'exécution :

```
[Spectateur] (j'entre en scene)
Quel age ai-je (>0)? 35
Combien d'argent ai-je en poche dans [0,99]? 112
Combien d'argent ai-je en poche dans [0,99]? 12
[Spectateur] (je suis la)
[Magicien] un petit tour de magie...
[Spectateur] (j'ecris le papier)
[Spectateur] (je montre le papier)
[Assistant] (je lis le papier)
[Assistant] (je calcule mentalement)
[Assistant] J'annonce : 3397 !
[Magicien]
 - humm... je vois que vous etes ages de 35 ans
   et que vous avez 12 euros en poche !
```