

# NFA032 : TP 3 (Objets partie 1 : étude de code)

13 mars 2018

## 1 Préliminaires

Lancez Eclipse et créez un nouveau projet de nom Tp3-032.

## 2 Exercice 1 : étude de la classe Compte

### 2.1 Création d'un paquetage `comptes`

Créez un paquetage de nom « `comptes` » (par convention les noms de paquetages sont en minuscules). Il contiendra tous les fichiers utilisés dans cet exercice.

### 2.2 Création de la classe `Compte`

Dans le paquetage `comptes` créez une classe de nom `Compte` et tapez dedans le code suivant :

---

```
public class Compte {
    private int numero;
    private double solde;

    public Compte(int num, double init){
        numero = num;
        solde = init;
    }
    public double getSolde() {
        return solde;
    }
    public int getNum() {
        return numero;
    }
    public void depot(double n){
        solde = solde+n;
    }
    public void retrait(double m) {
        solde = solde-m;
    }
}
```

---

Les questions suivantes vous permettront de contrôler votre compréhension du cours sur les objets et classes. Demandez de l'aide pendant les séances de Tp ou allez consulter les notes de cours si vous avez besoin d'éclaircissements.

1. Assurez vous que votre code compile : il ne doit pas y avoir de croix rouge ni de code sous-ligné en rouge dans votre fichier.
2. Notez que les variables d'instance sont déclarées *private*. Que peut-on faire ou ne pas faire avec des variables ou méthodes déclarées *public* ou *private* ?
3. Notez que la classe `Compte` possède un constructeur. A quoi reconnaît-on un constructeur dans une classe ?
4. A quoi sert un constructeur et comment est-il utilisé ?
5. Que fait le constructeur de cette classe une fois invoqué ?
6. A quoi sert la classe `Compte` ? Pourquoi ne possède-t-elle pas de méthode `main` ?

### 2.3 Création de la classe `TestComptes`

Toujours dans le paquetage `comptes` créez une nouvelle classe `TestComptes`, cette fois avec une méthode `main` que l'on va exécuter afin de créer et d'utiliser des objets instance de la classe `Compte`. Voici le code de la classe `TestComptes` :

```
1 public class TestComptes {  
2     public static void main(String[] args) {  
3         // Déclaration et création d'un premier objet Compte  
4         // On utilise le constructeur défini, qui initialise le numéro et le solde du compte.  
5         Compte c1 = new Compte(1, 100.00);  
6         // Affichons le solde courant de c1  
7         System.out.println(c1.getSolde());  
8         // Appel à la méthode depot sur c1  
9         c1.depot(55.35);  
10        // Appels aux méthodes getSolde() et getNum()  
11        System.out.println("Nouveau_solde_du_compte_numero_" + c1.getNum()+"_=" + c1.getSolde());  
12        // Enlevez le commentaire sur la ligne suivante et expliquez l'erreur.  
13        //double x = c1.solde;  
14    }  
15 }
```

1. Etudiez ce code et expliquez le comportement de chaque ligne du `main`.
2. Enlevez le commentaire ligne 13 : expliquez l'erreur qui apparaît. De quel genre d'erreur s'agit-il ? Comment donner à `x` la valeur courante du solde de `c1` ?
3. Exécutez ce programme (cliquez sur le bouton *Run*) on.
4. Ajoutez du code dans la méthode `main` afin de créer un deuxième compte `c2` sur lequel vous ferez un dépôt de 150 euros.
5. Ajoutez dans la classe `Compte` la méthode suivante.

```
public void affiche(){  
    System.out.print("Numero:_" + numero);  
    System.out.print("_solde:_" + solde);  
}
```

Expliquez à quoi sert cette méthode, et modifiez votre méthode main pour afficher les comptes c1 et c2 avant et après chaque dépôt **en invoquant cette méthode**.

## 2.4 Tableaux de Comptes, exécution avec PythonTutor

1. Créez une nouvelle classe `TesteTabComptes` contenant une méthode main qui crée un tableau de 5 comptes.
2. Dans une première boucle, créer et initialiser chaque compte avec 20 euros plus une somme égale à 200 euros multipliés par l'indice du compte dans le tableau.
3. Avec une nouvelle boucle, réaliser un retrait de 15 euros sur chaque compte, puis afficher le compte (en utilisant la méthode `affiche()`).

### 2.4.1 Exécuter avec PythonTutor

*PythonTutor* est une plateforme web qui vous permet d'exécuter des programmes écrits en plusieurs langages, tout en visualisant une représentation des objets et tableaux créés en mémoire. Rendez vous sur le site <http://www.pythontutor.com/> et cliquez sur `Start Visualizing your code now`.

1. Dans le menu déroulant tout en haut sélectionnez `Java 8` en tant que langage du programme à exécuter.
2. dans la fenêtre qui apparaît, enlevez la classe vide proposée et copiez-collez tout le code votre classe `Compte` **sauf la première ligne qui déclare le nom du package**.
3. Dans la fenêtre PythonTutor, placez vous dans la classe `Compte`, juste **avant sa dernière accolade**. Ajoutez la méthode main suivante :

---

```
public static void main(String[] args) {
    Compte c100 = new Compte(100, 100.00);
    c100.depot(55.35);
    // Création tableau
    Compte[] tc = new Compte [5];
    int numC = 10;
    // Création comptes
    for (int i=0; i<tc.length; i++) {
        tc[i]= new Compte(numC+i, 20+i*200);
    }
    // Retrait sur les comptes
    for (int i=0; i<tc.length; i++) {
        tc[i].retrait(15);
    }
}
```

---

4. Cliquez sur le bouton « Visualize execution ». Cela démarre l'exécution.
5. Pour continuer à exécuter, cliquez sur le bouton « *forward* ».

## 2.5 Virements et titulaires

1. Ajoutez dans la classe `Compte` la méthode `virerVers` donnée ci-dessous :

---

```
public void virerVers(double m, Compte dest){
    this.retrait(m);
    dest.depot(m);
}
```

---

2. Etudiez ce code et ajoutez un commentaire qui explique son comportement.
3. Dans le main de la classe `TestTabComptes`, ajoutez une boucle où chaque compte du tableau effectue un virement de 10 euros sur tous les autres comptes se trouvant à sa droite dans le tableau. Visualisez son exécution en utilisant `PythonTutor`.
4. Ajoutez dans la classe `Compte` une nouvelle variable qui correspond au nom du titulaire du compte. Vous devrez modifier plusieurs méthodes pour tenir compte de cette nouvelle donnée : (a) modifiez le constructeur de la classe pour qu'il initialise cette variable ; (b) modifiez la méthode d'affichage pour qu'elle l'affiche ; (c) ajoutez une méthode accesseur qui retourne le nom du titulaire.

### 3 A rendre (partie 1)

Vous rendrez un fichier zip avec le contenu de votre projet `Tp3`, qui doit contenir tous les fichiers demandés pour l'exercice 1. **Attention : vos fichiers doivent compiler, autrement votre devoir ne sera pas corrigé.**