



---

## TI215 : programmation orientée objet 1

### Exercices : série 4

---

#### Consignes

- Déclarez **private** les **variables d'instance** de toutes les classes. Prévoyez les **getters** et **setters** nécessaires (et uniquement ceux-là).
- Dessinez au fur et à mesure le **schéma des classes** afin d'avoir une vue d'ensemble de l'exercice.
- Chaque fois que vous avez créé une classe, testez-la ; prévoyez les instructions dans la méthode main pour, au minimum :
  - créer un objet de la classe ;
  - afficher sa description (toString( )) ;
  - appeler les méthodes que vous auriez ajoutées.

#### **4.1.** Créez un nouveau projet appelé **Immobilier**.

#### Objectif 1: Appliquer l'héritage

① Créez une classe intitulée **Localité** qui contient les variables d'instances suivantes :

*de type String :*

- nomLocalité

*de type int :*

- codePostal

Prévoyez un constructeur ainsi que la méthode **toString()** qui doit retourner la description de tout objet de cette classe en respectant le format suivant:  
*1000 Bruxelles.*

② Créez une classe intitulée **Personne** qui contient les variables d'instances suivantes :

*de type String :*

- nomPrénom

- téléphone

*de type char :*

- sexe

Prévoyez un filtre sur la variable sexe afin de n'accepter que les valeurs M, m, F ou f. Par défaut, à la création, si la variable est erronée, on mettra la valeur m.

Prévoyez un constructeur ainsi que la méthode **toString()** qui doit retourner la description de tout objet de cette classe en respectant le format suivant:

S'il s'agit d'une femme : **Madame Marie Legrand (tel : 0477/71.22.23)**

S'il s'agit d'un homme : **Monsieur Jules Petit (tel : 0473/21.72.73)**

③ Créez une classe intitulée **BienImmobilier** qui contient les variables d'instances suivantes :

de type *String* :

- rueNuméro (le numéro dans la rue et le nom de la rue de l'adresse)

de type **Localité** :

- localité (localité de l'adresse)

de type *int* :

- annéeConstruction (l'année de la construction du bâtiment)

de type **Personne** :

- propriétaire (personne propriétaire du bâtiment)

Prévoyez un constructeur.

Ajoutez les méthodes suivantes :

- une méthode appelée **tauxTVA** qui retourne le pourcentage de TVA à appliquer en cas de travaux : 6 % si la construction a plus de 20 ans, 21% sinon ;
- la méthode **toString()** qui doit retourner la description de tout objet de cette classe en respectant le format suivant:

**L'immeuble situé 100, Blvd du souverain à 1000 Bruxelles  
appartenant à Monsieur Jules Petit (tel : 0473/21.72.73)**

Attention, vous devez écrire le moins possible de code. Par conséquent, faites (implicitement) appel le plus possible aux méthodes *toString()* des classes référencées (ici, *Localité* et *Personne*) !

④ Créez une classe intitulée **Immeuble** qui est une **sous-classe de BienImmobilier** et qui contient, en plus, les variables d'instances suivantes :

de type *int* :

- nbStudios (nombre de studios dans l'immeuble)
- nbAppartements (nombre d'appartements dans l'immeuble)

de type double :

- loyersStudios (somme des loyers mensuels perçus dans l'ensemble des studios)
- loyersAppartements (somme des loyers mensuels perçus dans l'ensemble des appartements)

Prévoyez un constructeur.

Ajoutez les méthodes suivantes :

- la méthode appelée **revenuMensuel** qui retourne le montant total des loyers mensuels perçus dans l'immeuble ;
- la méthode **toString()** qui doit retourner la description de tout objet de cette classe en respectant le format suivant:

*L'immeuble situé 32, rue du bouchon à 4000 Liège  
appartenant à Monsieur Pierre Duvin (tel : 0472/91.92.98)  
contient 5 studio(s) et 6 appartement(s) et rapporte 8800 euros  
par mois*

Dans la méthode main de la classe Principal, affichez le taux de TVA applicable en cas de travaux pour chaque immeuble créé.

## **Objectif 2 : Accéder à des variables d'instance héritées déclarées private**

⑤ Créez une classe intitulée **Villa** qui est une **sous-classe de BienImmobilier** et qui contient en plus les variables d'instances suivantes :

de type int :

- nbChambres (nombre de chambres de la villa)

de type double :

- superficieTerrain (nombre d'ares que compte le terrain sur lequel est construite la villa)

de type boolean :

- campagne (vrai si la villa est située à la campagne)

Prévoyez un constructeur ainsi que la méthode **toString()** qui doit retourner la description de tout objet de cette classe en respectant le format suivant:

*L'immeuble situé 1, rue Bohème à 5098 Nil-la-campagne  
appartenant à Madame Andrée Petit (tel: 0471/71.72.73 )  
est une villa à 5 chambre(s) située sur un terrain de 15.5 ares*

Ajoutez la méthode **descriptionTerrain** qui retourne la chaîne de caractères respectant le format suivant :

*Terrain de 15.5 ares situé à la campagne, 1, rue Bohème à Nil-la-campagne*

N.B. Remplacer "à la campagne" par "en ville" si le booléen *campagne* est à faux.

© Créez une classe intitulée **VillaLouée** qui est une **sous-classe de Villa** et qui contient en plus les variables d'instances suivantes :

de type **Personne** :

- locataire (personne qui loue la villa)

de type *double* :

- loyer (montant du loyer mensuel que paie le locataire)

de type *int* :

- duréeBail (durée du bail en nombre de mois)

Prévoyez un constructeur.

Ajoutez les méthodes suivantes :

- la méthode **toString()** qui doit retourner la description de tout objet de cette classe en respectant le format suivant:

*L'immeuble situé 50, rue de la fontaine à 5000 Namur  
appartenant à Madame Marie Legrand (tel: 0471/71.22.23 )  
est une villa à 4 chambre(s) située sur un terrain de 15 ares  
et est louée à Monsieur Jean Larent (tel: 0477/71.72.73 )  
pour un loyer de 500.50 euros par mois (durée du bail: 3 an(s))*

- la méthode appelée **revenuPropriétaire** qui retourne la chaîne de caractères décrivant le revenu annuel que touche le propriétaire en respectant le format suivant:

*Marie Legrand touche un loyer de 6000 euros par an*

### **Objectif 3 : Comprendre le polymorphisme**

⑦ Créez une classe intitulée **ImmeubleBureaux** (immeuble à studios, appartements et bureaux) qui est une **sous-classe de Immeuble** et qui contient en plus les variables d'instances suivantes :

de type *int* :

- nbM2 (nombre de m<sup>2</sup> de bureaux que comprend l'immeuble)

de type *double* :

- prixM2 (prix de la location au m<sup>2</sup>)

Prévoyez un constructeur.

Ajoutez les méthodes suivantes :

- une méthode appelée **revenuMensuel** qui retourne le revenu mensuel total de l'immeuble (total des loyers + revenu de la location des bureaux) ;
- la méthode **toString( )** qui doit retourner la description de tout objet de cette classe en respectant le format suivant:

*L'immeuble situé 105, Avenue Louise Legrand à 5000 Namur  
appartenant à Madame Fabienne Lesire (tel: 0475/71.72.73 )  
contient 3 studio(s) et 2 appartement(s) et rapporte 11000 euros  
par mois  
(il comprend 100 m2 de bureau dont la location rapporte 5000  
euros)*

*Soyez particulièrement attentifs à la description ainsi affichée des objets de type ImmeubleBureau. Il y a polymorphisme : la méthode revenuMensuel qui a été appelée est la méthode la plus spécifique, celle de la classe ImmeubleBureaux et pas la méthode revenuMensuel de la classe Immeuble. A savoir, les 11000 euros affichés correspondent à la somme des loyers perçus pour la location des studios et appartements de l'immeuble (6000 euros) et des revenus de la location des bureaux (5000 euros).*

⑧ Créez une classe intitulée **ImmeubleRezCommercial** (immeuble à rez-de-chaussée commercial, càd immeuble à studios, appartements et magasins) qui est une **sous-classe de Immeuble** et qui contient en plus les variables d'instances suivantes :

de type *int* :

- nbMagasins (nombre de magasins loués au rez-de-chaussée de l'immeuble)

de type double:

- revenuLocationMagasins (revenu total de la location de l'ensemble des magasins)

Prévoyez un constructeur.

Ajoutez les méthodes suivantes :

- une méthode appelée **revenuMensuel** qui retourne le revenu mensuel total de l'immeuble (total des loyers + revenu de la location des magasins) ;
- la méthode **toString( )** qui doit retourner la description de tout objet de cette classe en respectant le format suivant:

*L'immeuble situé 77, rue du Prince à 6000 Charleroi  
appartenant à Monsieur Jacques Dutronc (tel: 0499/51.32.73)  
contient 11 studio(s) et 4 appartement(s) et rapporte 32000  
euros par mois  
(il comprend un rez de chaussée commercial composé de 2  
magasins dont le revenu commercial est de 10000 euros)*

Soyez particulièrement attentifs à la description ainsi affichée des objets de type *ImmeubleRezCommercial*. Il y a **polymorphisme** : la méthode *revenuMensuel* qui a été appelée est la méthode **la plus spécifique** (et pas la méthode *revenuMensuel* de la classe *Immeuble*). A savoir, les **32000 euros affichés correspondent à la somme des loyers perçus pour la location des studios et appartements de l'immeuble (22000 euros) et des revenus de la location des magasins (10000 euros).**

⑨ Dans la méthode main de la classe Principal :

- affichez le taux de TVA applicable en cas de travaux pour chacune des villas créées ;
- affichez le code postal de la localité où se situe chacun des immeubles créés (immeubles, immeubles à bureaux, immeubles à rez-de-chaussée commercial) ;
- affichez le nom et le prénom du propriétaire de chacun des immeubles à bureaux, immeubles à rez-de-chaussée commercial et des villas (louées ou non).

## 4.2. STATIC

### Objectif 4: Manipuler les variables et méthodes de classe

Prévoyez les **variables et méthodes de classe** nécessaires pour pouvoir afficher dans la classe Principal les informations suivantes :

- ① Le nombre d'immeubles à rez-de-chaussée commercial créés par le programme principal en ne tenant compte que des immeubles comprenant trois magasins ou plus.
- ② Le nombre de villas **louées** créées par le programme principal dont la superficie du terrain est supérieure à 10 ares.
- ③ Le nombre d'immeubles créés par le programme principal appartenant à « Richard Mateur ».
- ④ Le nombre moyen d'appartements que contiennent tous les immeubles créés par le programme principal.

*Attention, ces variables statiques doivent être déclarées avec la protection **private**. A vous de prévoir les getters publics correspondant ! Ceux-ci doivent être déclarés **static** puisqu'ils permettent d'accéder à des caractéristiques de classes elles-mêmes déclarées statiques.*