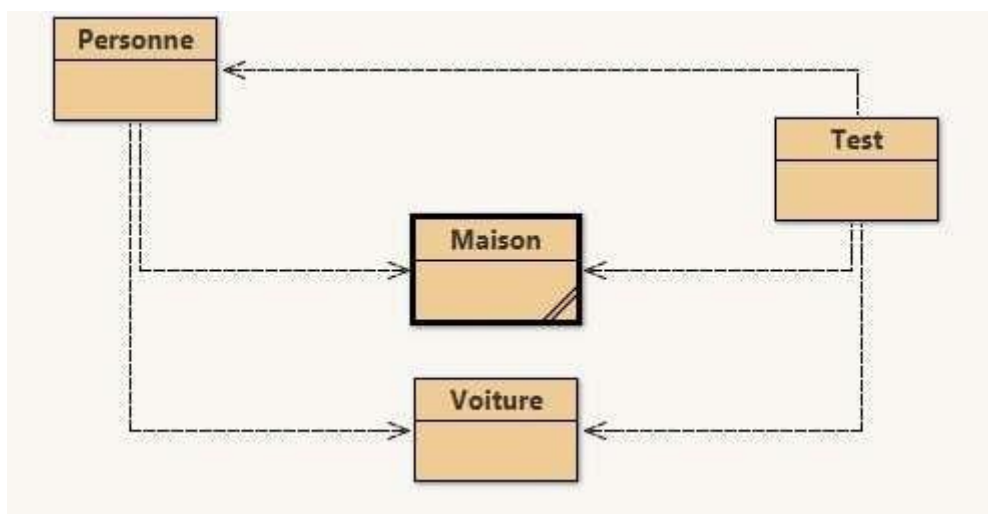


TP02

Exercice 1 : Personne et possessions

Cet exercice comporte 3 classes différentes auxquelles vient s'ajouter une classe Test (fichier java fourni sur cyberlearn à rajouter dans un projet que vous créez !). La classe Test vous permettra, comme son nom l'indique, de tester vos classes en créant des instances. Vous n'êtes pas autorisé à modifier cette classe. Vous êtes en revanche encouragés à regarder les différentes signatures (noms) des méthodes appelées afin que vous puissiez les créer. Ne vous inquiétez donc pas si la classe Test est détectée comme comportant des erreurs. C'est à vous de créer les classes manquantes pour les résoudre ;)



La classe personne possède les attributs suivants :

- Nom
- Prénom
- Adresse
- Age
- Voiture
- Maison

A noter que pour les deux derniers attributs, il s'agit d'une instance issue des classes Maison et Voiture respectivement.

Une instance de Personne doit pouvoir être créée de 4 manières différentes :

- Sans voiture ni maison (vit chez ses parents)
- Avec une voiture sans maison (vit chez ses parents)
- Sans voiture avec une maison
- Avec une voiture et avec une maison

Lorsqu'un de ces deux attributs n'est pas utilisé dans le constructeur, il faut mettre sa valeur à `null`

Il doit être possible au moyen de méthodes (fonctions) de changer sur une instance la voiture ou la maison d'une personne.

Une méthode `afficher()` doit être implémentée afin de gérer l’affichage d’une instance de `Personne`. Si la personne n’a pas de possessions, un message l’indique (exemple de sortie).

De plus, une méthode doit être implémentée afin de pouvoir afficher le prénom de la première personne enregistrée (voir exemple de sortie).

La classe `Maison` comporte les attributs suivants :

- `Surface`
- `NbPièces`
- `Location` (boolean)
- `Prix`

Selon si la maison est sous location ou non, l’affichage du prix sera différent (voir exemple de sorties)

La classe `Voiture` comporte les attributs suivants :

- `Marque`
- `Modèle`
- `Année`

```
Prénom de la première personne enregistrée : Paul
```

```
Affichage des personnes
```

```
-----
```

```
PasDupont Paul(24), Chemin de la route 35
```

```
Possessions(s) : Ford Fiesta de 2019.
```

```
Grosjean Felix(37), Chemin de la rue 97
```

```
Possessions(s) : Maison d'une surface de 200m2 comportant 8 pièces. Le loyer est de 7050Chf mensuel.
```

```
Harat Séverine(58), Chemin de l'impasse 12
```

```
Possessions(s) : Porsche Cayenne de 2007. Maison d'une surface de 600m2 comportant 14 pièces. Le loyer est de 10800Chf mensuel.
```

```
Carat Aude(19), Chemin du sentier 34
```

```
Possessions(s) : Aucunes
```

```
Camar Jack(44), Chemin de la route 35
```

```
Possessions(s) : Audi S3 de 2022. Maison d'une surface de 100m2 comportant 6 pièces. Le prix est de 1600000Chf.
```

Exercice 2 : Gestion d'une école

L'objectif de cet exercice est de créer une classe « école » afin de mettre en pratique vos connaissances fraîchement acquises sur les classes, la méthode toString et la méthode equals. Une classe Main est fournie pour vous permettre de tester votre code.

Vous aurez en plus de cela une deuxième classe à mettre en place permettant de correctement assimiler les notions de l'exercice 1 dans laquelle vous allez créer une « Personne », classe où vous stockerez l'école dans laquelle la personne suit ses études.

Enoncé

Voici le détail de la classe représentant une école. Une école possède les champs suivants :

- Nom
- Adresse
- Type (Publique / Privée)
- Nombre de bâtiments
- Degré (Primaire, C.O. ou Universitaire)
- Domaine (Général, Informatique, Gestion, ...)
- Nombre d'étudiants

Cette classe doit posséder les méthodes suivantes :

- Changer le nombre de bâtiments par une valeur donnée
- Changer le nombre d'étudiants par une valeur donnée
- Une méthode toString affichant l'instance comme dans l'exemple
- Une méthode equals()
 - Si les deux écoles comparées sont privées, les attributs comparés seront
 - Nom
 - Adresse
 - Domaine
 - Sinon
 - Nom
 - Adresse
- Changer le degré par une valeur donnée :
 - Si l'école est publique
 - Si le degré est Primaire ou C.O., le domaine est obligatoirement Général

Affichage des écoles :

Haute école de gestion, Route de drize 19

Ecole Publique de degré Tertiaire dans le domaine Gestion

Il y a 5 bâtiments pour 1500 étudiants

Schultz, Route du lac 29

Ecole Privée de degré Secondaire dans le domaine Economie

Il y a 2 bâtiments pour 200 étudiants

Collège Saint-Louis, Route de Corzier 23

Ecole Privée de degré Secondaire dans le domaine Général

Il y a 3 bâtiments pour 400 étudiants

Collège Saint-Louis, Route de Corzier 23

Ecole Privée de degré Secondaire dans le domaine Mathématiques

Il y a 1 bâtiments pour 50 étudiants

Collège de Carouge, Route des centenaires 34

Ecole Publique de degré Secondaire dans le domaine Général

Il y a 4 bâtiments pour 1500 étudiants

Collège de Carouge, Route des centenaires 34

Ecole Publique de degré Secondaire dans le domaine Général

Il y a 4 bâtiments pour 500 étudiants

Test des égalités entre écoles

Collège Saint-Louis, Route de Corzier 23

Ecole Privée de degré Secondaire dans le domaine Général

Il y a 3 bâtiments pour 400 étudiants

----- n'est pas égal à -----

Collège Saint-Louis, Route de Corzier 23

Ecole Privée de degré Secondaire dans le domaine Mathématiques

Il y a 1 bâtiments pour 50 étudiants

Collège de Carouge, Route des centenaires 34

Ecole Publique de degré Secondaire dans le domaine Général

Il y a 4 bâtiments pour 1500 étudiants

----- est égal à -----

Collège de Carouge, Route des centenaires 34

Ecole Publique de degré Secondaire dans le domaine Général

Il y a 4 bâtiments pour 500 étudiants

Concernant la classe « Personne », vous êtes libre d'ajouter des attributs de bases de votre choix (par exemple : nom, prenom, age, etc...), mais il faudra absolument un attribut « ecole » permettant de disposer des infos de l'école :

Cette classe doit posséder les méthodes suivantes :

- Récupérer le nom de l'école et l'afficher
- Afficher l'instance de l'école de l'étudiant comme vu dans l'exemple sur l'image.
- Une méthode toString qui affiche les infos de base de l'étudiant ainsi que les informations concernant son école selon les cas suivants :
 - S'il s'agit d'une école publique, il faut afficher les informations de l'école comme vu dans l'exemple précédemment.
 - Dans le cas où il s'agit d'un étudiant universitaire, rajoutez une ligne en plus avant d'afficher les informations de l'école précisant qu'il s'agit d'un étudiant dans le niveau tertiaire.
 - S'il s'agit d'une école privée, il suffit de préciser que l'école est privée et afficher le nom de l'école.

Avant de vous lancer dans cette deuxième partie de cet exercice, prenez le temps de réfléchir au code déjà réalisée qui pourra être réutilisé dans vos différentes méthodes ou à du code que vous rajouterez et qui pourrait aussi être réutilisé.

Pour tester votre code, vous pouvez modifier la classe Main fournie et rajoutez une partie où vous créez vos différentes instances de Personne et où vous testez les différentes méthodes à mettre en place.