

## Programmation Objet Avancée - TP5

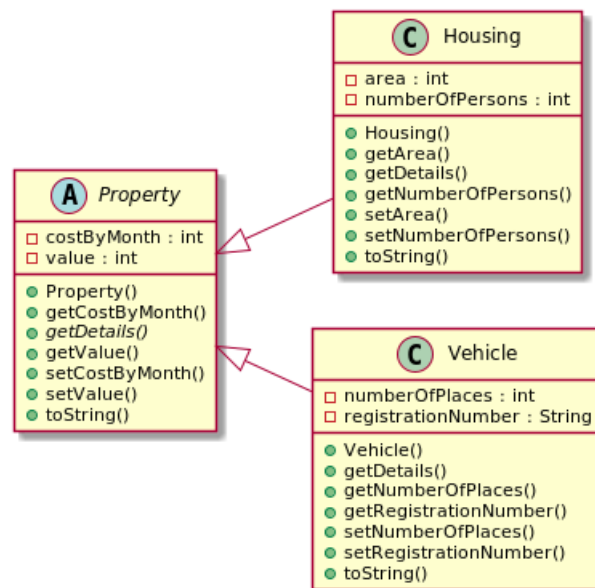
—o000o—o000o—

### Héritage et abstraction

## 1 Gestion d'une ville

Dans un souci de transparence envers ses citoyens, une ville décide de concevoir un système informatique lui permettant de représenter l'ensemble des biens (bâtiment, véhicule, . . .) qu'elle possède.

1. Un bien est défini par sa **valeur** et son **coût d'entretien mensuel**. Écrire la classe `Property`
2. Chaque bien appartient à une catégorie qui lui permet de posséder des informations supplémentaires. Ainsi, les **véhicules** possèdent un **numéro d'immatriculation** et un **nombre de places maximum**, tandis que les **logements** sont caractérisés par leur **superficie au sol** et le **nombre de personnes qu'ils peuvent accueillir**. Écrire les classes permettant de modéliser cette application



3. Redéfinissez la méthode `String toString()` dans l'ensemble des classes
4. Nous souhaitons rendre impossible l'instanciation d'un bien sans que son type ne soit précisé. Déclarez `Property` en tant que classe abstraite, puis définissez la méthode `String getDetails()`, (chaque détail étant spécifique à un type)
5. Pour le système d'information que nous sommes en train de développer, une ville peut être considérée comme une classe d'objets qui référence l'ensemble des biens qu'elle possède. Cette classe doit offrir les services suivants :
  - Consulter les informations d'un bien particulier
  - Consulter le nombre total de véhicules
  - Calculer le coût total mensuel d'entretien des biens
  - Consulter le nombre total de personnes logées
  - Calculer le coût total mensuel d'entretien pour l'ensemble des véhicules

\*\*\* **Pour aller plus loin**, vous pouvez utiliser les `stream()` offert par Java 8 pour implémenter les méthodes ci-dessus en se basant sur le lien suivant: <https://blog.axopen.com/2014/05/java-8-api-stream-introduction-collections/>. La figure ci dessous décrit un stream qui est considéré comme une séquence d'éléments sur laquelle on peut effectuer un groupe d'opérations de manière séquentielle ou parallèle ayant une source et plusieurs opérations à traiter.

