

TP1 : Rappel de notions de programmation objet

Table des matières

Exercice 1 – Location de voitures.....	2
Question 5 :.....	2
Question 6 :.....	2
Question 7 :.....	2
Question 11 :.....	3

Exercice 1 – Location de voitures

Question 5 :

- L'interface **Stream** permet de traiter les collections de manière séquentiel, en utilisant des opérations séquentielles et parallèles
- Pour obtenir un stream depuis une liste il faut utiliser la syntax: **liste.stream()**.
- La méthode filter retourne un stream. Elle permet de filtrer les éléments en fonction du prédicat donnée.
- La méthode map retourne aussi un stream. Elle consiste modifier chaque éléments du stream en appliquant la fonction donnée.
- On peut utiliser **Collectors.joining()** de la façons suivante :

```
@Override
public String toString() {
    return transports.stream()
        .map(Transport::toString)
        .collect(Collectors.joining("\n"));
}
```

Cela permet notamment de ne pas avoir le dernier saut à la ligne.

Question 6 :

- La méthode appelée par **CarRental.remove()** est la méthode **equals()**.

Question 7 :

- Si il n'y a pas de voiture correspondant à l'année demandée il faut renvoyer une liste vide.

Question 10 :

- Si **Car** et **Camel** étaient des classes nous aurions pu implémenter une classe abstraite.

Question 11 :

- La méthode **findACarByModel** doit renvoyer un **Optional** dans le cas ou il n'y a aucune voiture correspondant au model indiqué.