

### Exercice 3: Gestion de voitures

Reprenons la classe **Voiture** et ajoutons un système de Location des Voitures. Dans ce système, il existe un catalogue général qui gère la liste de toutes les voitures. Le catalogue général doit être implémenté à partir d'une classe nommée **InventaireVoiture**. Cette classe doit permettre d'enregistrer des nouvelles voitures dans le système. Le nombre maximum de voitures est limité à 100. La classe doit également permettre d'ajouter le kilométrage d'une voiture spécifique (spécifiée par son index dans le tableau) ainsi que d'afficher l'information de toutes les voitures dans le système ou d'un modèle de voiture spécifique.

1. Identifiez les *attributs*, les *constants* et les *méthodes* de la classe **InventaireVoiture**.

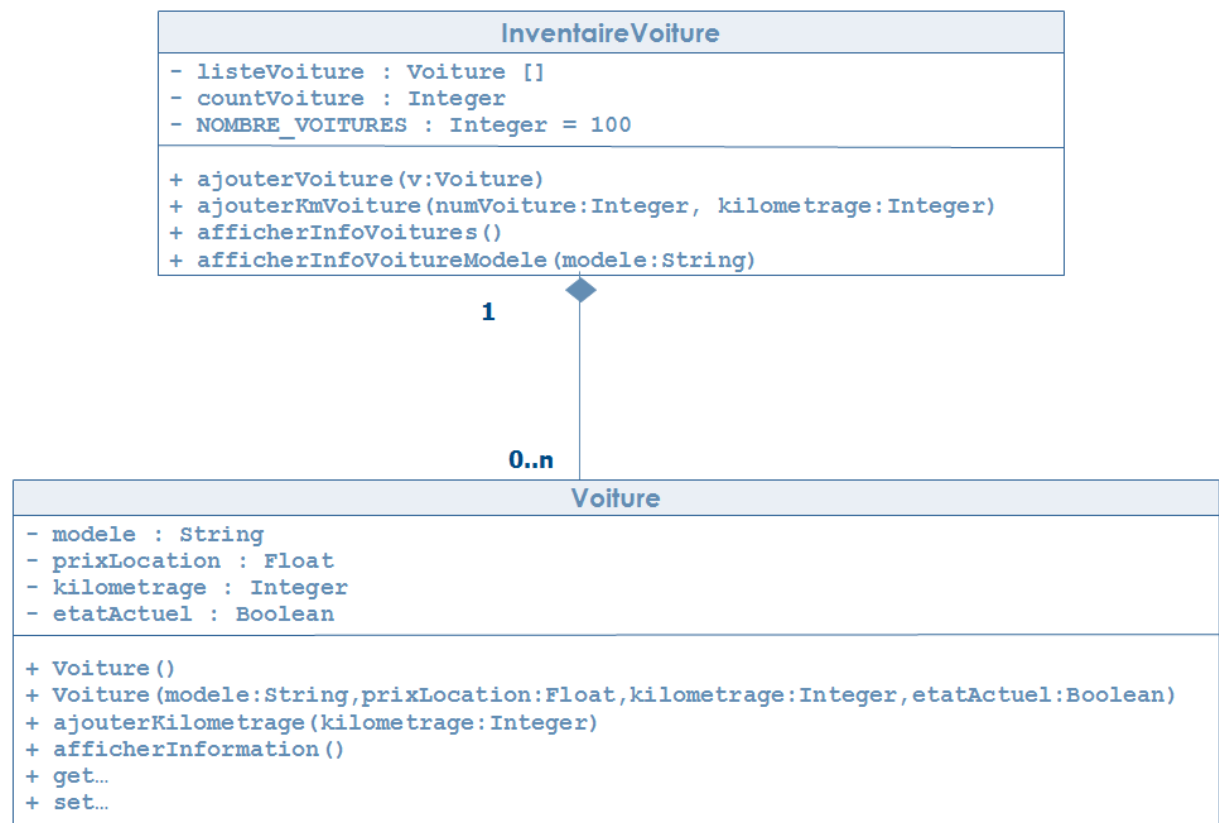
**Attributs** : listeVoiture, countVoiture

**Constants** : NOMBRE\_VOITURES = 100

**Méthodes** : ajouterVoiture, ajouterKmVoiture, afficherInfoVoitures, afficherInfoVoitureModele

2. Dessinez le diagramme UML de la classe **InventaireVoiture**. Quelle type de relation doit être liée entre la classe **InventaireVoiture** et la classe **Voiture**?

**Relation de composition**



3. A partir du diagramme UML, programmez en Java la classe **InventaireVoiture**.

```

public class InventaireVoiture {
    private Voiture[] listeVoiture;
    private int countVoiture;
    private final static int NOMBRE_VOITURES = 100;

    public InventaireVoiture ()
    {
        listeVoiture = new Voiture[NOMBRE_VOITURES];
    }
}
    
```

```

        countVoiture = 0;
    }

    public void ajouterVoiture(Voiture v)
    {
        if(countVoiture < NOMBRE_VOITURES)
        {
            listeVoiture[countVoiture] = v;
            countVoiture++;
        }
        else
            System.out.println("Erreur : l'inventaire est plein");
    }

    public void ajouterKmVoiture(int numVoiture, int kilometrage)
    {
        listeVoiture[numVoiture].ajouterKilometrage(kilometrage);
    }

    public void afficherInfoVoitures()
    {
        for(int i=0; i<countVoiture; i++)
        {
            listeVoiture[i].afficherInformation();
        }
    }

    public void afficherInfoVoitureModele(String modele)
    {
        for(int i=0; i<countVoiture; i++)
        {
            if(listeVoiture[i].getModele().compareTo(modele)==0)
                listeVoiture[i].afficherInformation();
        }
    }
}

```

4. Créer une nouvelle classe appelée **Programme** qui contiendra une méthode main. Dans la méthode main réaliser les actions suivantes :
  - a. Créez un objet de type **InventaireVoiture**.
  - b. Ajoutez les voitures suivantes dans le système :
    - Voiture 1 : Renault (modèle), 80.0 euros, 100 km, Louée
    - Voiture 2 : Volkswagen (modèle), 20.0 euros, 500 km, Pas louée
    - Voiture 3 : Volkswagen (modèle), 45.0 euros, 200 km, Louée
  - c. Ajoutez 20 km à la voiture d'indice 1.
  - d. En parcourant tout le tableau, Affichez l'information de toutes les voitures du modèle « Volkswagen »

```

public class Programme {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

        InventaireVoiture inv = new InventaireVoiture();
    }
}

```

```

    Voiture v1 = new Voiture("Renault", 80, 100, true);
    Voiture v2 = new Voiture("Volkswagen", 20, 500, false);
    Voiture v3 = new Voiture("Volkswagen", 45, 200, true);

    inv.ajouterVoiture(v1);
    inv.ajouterVoiture(v2);
    inv.ajouterVoiture(v3);

    inv.ajouterKmVoiture(1, 20);

    inv.afficherInfoVoitureModele("Volkswagen");
}
}

```