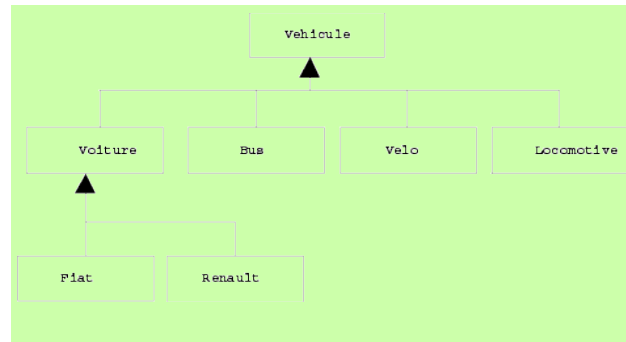


## TD VROOUUMM !!!

Soit la hiérarchie de classes suivante:



Soit la classe `Vehicule` suivante:

```
public class Vehicule{
    protected int nbRoues;

    public void demarrer(){};
    public void rouler(){};
    public void arreter(){};

    public String categorie(){
        return "Je suis un vehicule";
    }

    public void description(){
        System.out.println(categorie() + " a " + nbRoues + " roues");
    }
}
```

### Question 1

Écrire les classes `Voiture`, `Bus` et `Fiat`. Dans ces 3 classes, le nombre de roues sera spécifié (4 pour les voitures et 6 pour les bus). Ces classes redéfinissent la méthode `categorie()` pour que la méthode `description()` donne des messages différents (« je suis une voiture a 4 roue », « je suis un bus à 6 roues », « je suis une fiat à 4 roues »).

Soit la classe `Client` contenant la méthode suivante :

```
public static void main(String args[]){
    Vehicule v;
    v = new Vehicule();
    v.description();
    v = new Voiture();
    v.description();
    v = new Bus();
    v.description();
    v = new Fiat();
    v.description();
}
```

Que produit ce code ? Quels sont les mécanismes mis en jeu ?

### Question 2

Dans la classe `Vehicule`, que se passe-t-il si on protège (par `private`) la méthode `categorie()` ?

### Question 3

Écrire une interface `Avertir` déclarant une méthode `klaxoner()`. Tous les `Vehicule` implémentent cette interface. Lorsque la méthode `klaxoner()` est invoquée sur une `Voiture` elle affiche « Pouet Pouet », lorsqu'elle est appelée sur un `Bus` elle affiche « Tut Tut ». Faites-le nécessaire.