

Ilias SAYAGH

**Couplage faible**

Matière **: JEE**

**Pour le 20 mars 2023**

:   
  
  
  
sous la direction de Mme Tijane BADRI

**Année universitaire 2022-2023**

**Introduction**

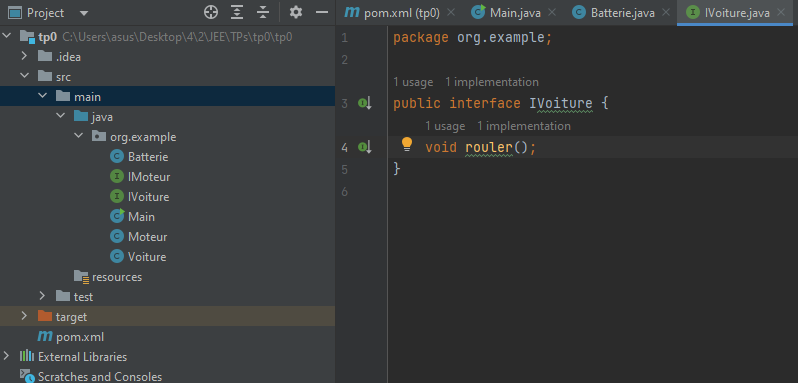
Un couplage fort entre les classes est une liaison direct qui restreint l’adaptabilité du programme si le besoin viens à s’améliorer, d’où l’utilisation du couplage faible qui consiste à introduire le principe d’interface rendant ainsi notre programme flexible pour toute amélioration.

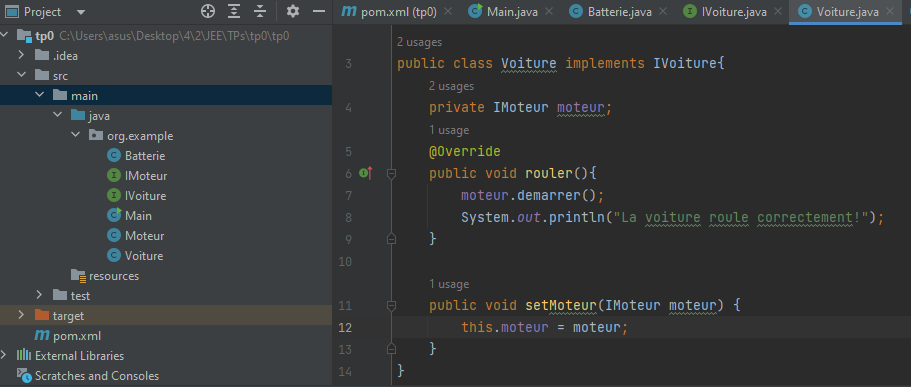
Cette technique nous permet au même temps de respecter les principes majeurs de conception qui consiste à rendre notre application ouverte à l’extension pour permettre notre application d’évoluer et de s’adapter au changement.

Le programme réalisé simulent le cas où on voudrait ajouter une nouvelle catégorie de voiture celle avec une batterie au lieu d’un moteur, nous pourrons ainsi voir l’utilité du couplage faible dans un exemple concret.

**Réalisation**

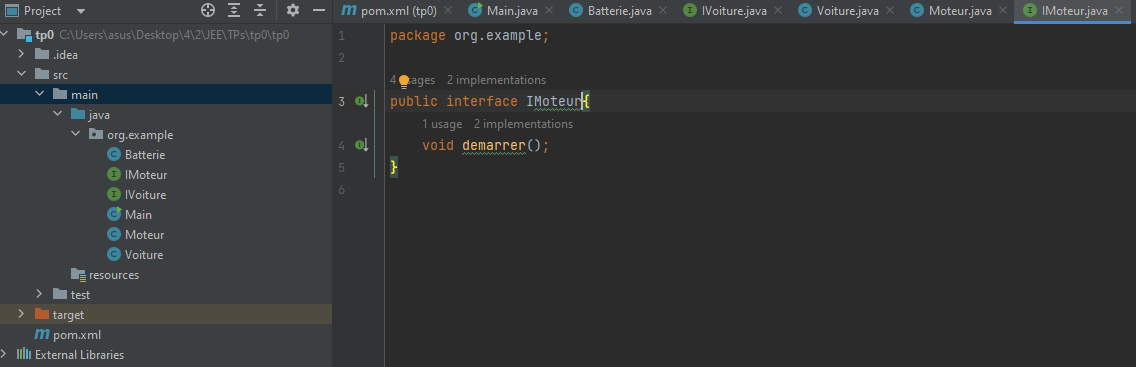
**1/ Interface voiture**

****

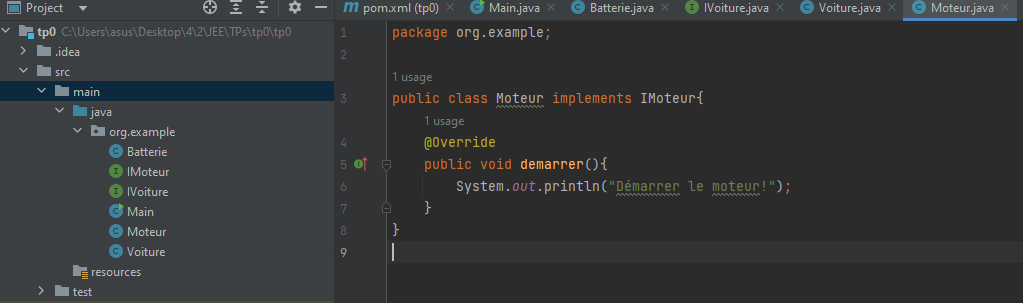
**2/ Class voiture**

L’utilisation de l’interface IMoteur nous permettra ensuite de choisir librement dans le programme principal (main) le type de voiture soit avec moteur ou avec batterie.

**3/ Interface du Moteur**

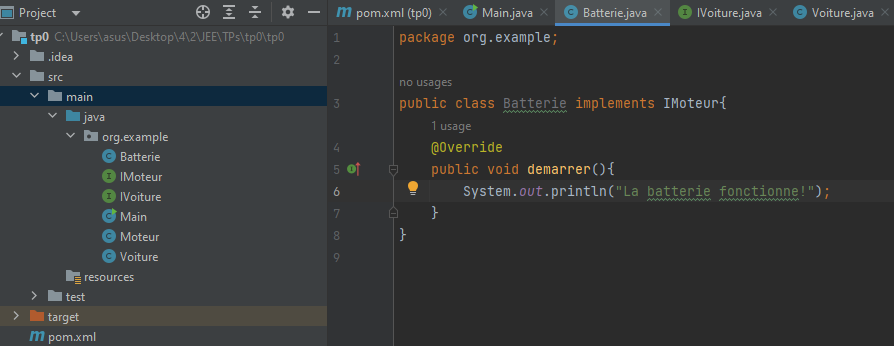
****

**4/ Class Moteur**

****

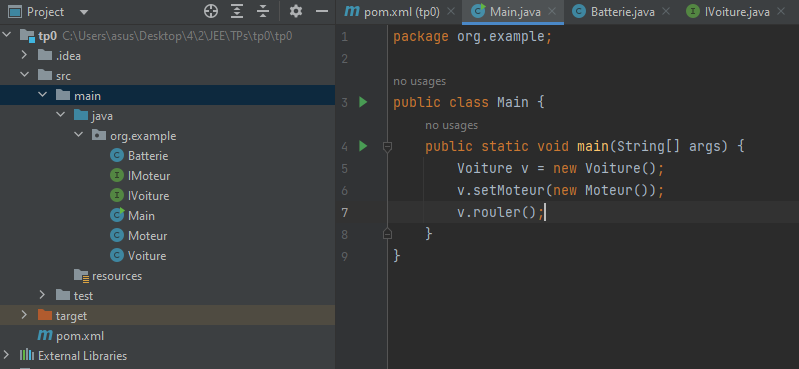
La class Moteur contient une fonction qui permet de démarrer le moteur.

**5/ Class Batterie**

****

La class Batterie contient une fonction qui indique que la batterie fonctionne.

**6/ Main**



Le programme principal permet de faire rouler une voiture quel soit avec un moteur en spécifiant dans les paramètres de la méthode setMoteur() « new Moteur() » ou avec une batterie en spécifiant new Batterie().

**Conclusion**

En guise de conclusion pour donner la faculté d’adaptabilité à son code il faut intégrer le principe d’interface permettant ainsi d’utiliser différente classe pour les même fonctionnalités.