

# Banque 02

## Interface

### 1 Suite du projet

#### Interface ICompteRemunere

Toujours dans le ProjetBank, créez une nouvelle interface **ICompteRemunere** qui respectera les contraintes suivantes :

- Méthodes (au moins) :
  - **public double calculerInterets()**
  - **public void verserInterets()**
  - **public double getTaux()**
  - **public void setTaux(double unTaux)**

Votre classe CompteRemunere doit implémenter cette interface

#### Interface ICompteASeuil

Toujours dans le ProjetBank, créez une nouvelle interface **ICompteASeuil** qui respectera les contraintes suivantes :

- Méthodes (au moins) :
  - **public void retirer(double uneValeur)**
  - **public double getSeuil()**
  - **public void setSeuil(double unSeuil)**

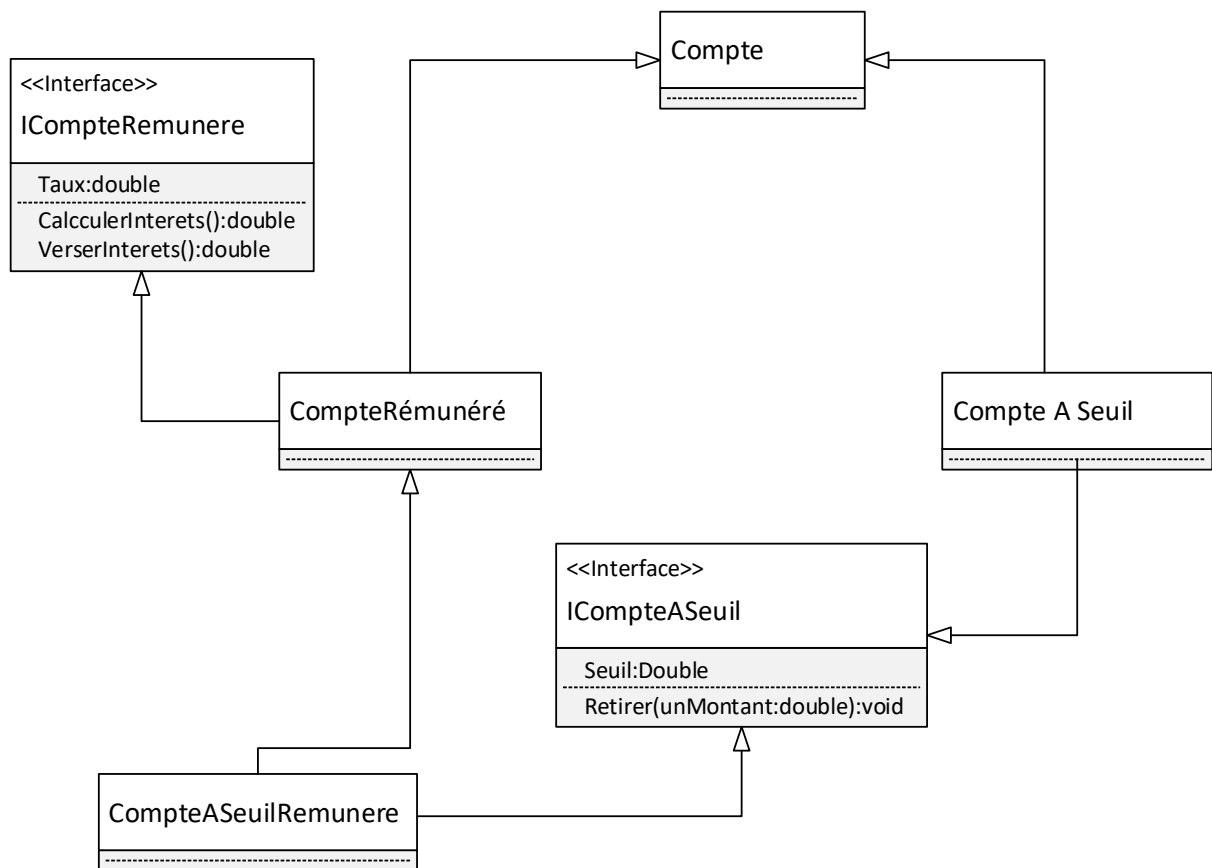
Votre classe CompteASeuil doit implémenter cette interface

#### Classe CompteASeuilRemunere

Toujours dans le ProjetBank, créez une nouvelle classe **CompteASeuilRemunere** qui respectera les contraintes suivantes :

- Cette classe hérite de la classe CompteRemunere **ou** CompteASeuil. Il vous appartient de décider.
- Cette classe implémente ICompteRemunere **et/ou** ICompteASeuil. Il vous appartient de décider. Le choix sera dépendant du choix de votre classe parent
- Attributs : seuil ou taux selon votre choix d'héritage
- Constructeurs : aux choix
- Méthodes en fonction de l'héritage.

Par exemple, ici on a choisit d'hériter de CompteRemunere



## Test

Dans votre classe Program / Run complétez votre méthode main, ajoutez des **CompteASeuilRemunere**.

Modifiez ou vérifiez que votre code qui permet de verser les intérêts sur tous les comptes de type rémunéré fonctionne aussi avec les **CompteASeuilRemunere** (faites usage de l'interface **ICompteRemunere** au lieu de la classe)