

-----

Cette feuille poursuit le développement du package ***\_Banque***, avec deux nouvelles classes : ***OperationBancaire*** et ***\_Banque***.

**Exercice      Module de servitudes**

Analyser soigneusement les codes sources fournis en annexe 1.

Développer et exécuter avec succès un module de tests unitaires correspondant.

**Exercice      La classe *OperationBancaire* - Version 1.0.0**

On considère la spécification technique détaillée de la classe ***OperationBancaire*** et le fichier "header" fournis en annexe.

Etudier en détail le fichier "header" fourni et contrôler sa conformité vis à vis des spécifications de la classe.

**Exercice      La classe *OperationBancaire* - Version 1.0.0**

Développer les constructeurs et le destructeur de la classe cible.

Exécuter avec succès le module de tests unitaires correspondant, fourni en annexe.

**Exercice      La classe *OperationBancaire* - Version 1.0.0**

Développer les autres codes opérationnels de cette classe.

Exécuter avec succès tous les modules de tests unitaires fournis.

**Exercice      La classe *OperationBancaire* - Version 1.0.0**

Développer et exécuter avec succès un module de tests unitaires des cas d'anomalies.

**Exercice      La classe abstraite *\_Banque* - Version 1.0.0**

Centraliser dans une nouvelle classe ***\_Banque*** :

- les directives d'inclusion nécessaires aux classes ***Titulaire*** et ***OperationBancaire***,
- les fonctions de servitudes nécessaires à chacune de ces classes,
- tous les littéraux exploités par ces classes, à l'exception de l'identifiant de chacune d'elles.

Les fonctions de servitudes doivent devenir des méthodes de la classe.

Interdire la création d'objets, instances de la classe ***\_Banque***.

Rassembler dans cette nouvelle classe tous les modules de tests unitaires correspondants et exécuter chacun d'eux avec succès.

**Exercice      La classe abstraite *\_Banque* - Version 1.0.0**

Compléter les modules de tests unitaires de la classe ***\_Banque***.

**Exercice**      **La classe *OperationBancaire* - Version 1.1.0**

Modifier la classe existante *OperationBancaire* pour prendre en compte la mise en place de la classe *\_Banque*.

Exécuter avec succès tous les modules de tests unitaires existants.

**Exercice**      **La classe *Titulaire* - Version 1.1.0**

Modifier la classe existante *Titulaire* pour prendre en compte la mise en place de la classe *\_Banque*. Exécuter avec succès tous les modules de tests unitaires existants.

**Exercice**      **La classe *Tests* - Version 1.0.0**

Analyser tous les codes sources de la classe *Tests*.

- Q1**    Recenser les attributs d'instances
- Q2**    Recenser les attributs de classe
- Q3**    Recenser les méthodes d'instance
- Q4**    Recenser les méthodes de classe
- Q5**    Justifier la répétition de la déclaration des attributs de classe en tête du module *Tests.cpp*

Pourrait on diminuer le nombre de méthodes de classe (notamment privées) en les remplaçant par des méthodes d'instances ?

**Exercice**      **La classe abstraite *\_Banque* - Version 1.1.0**

Doter la classe *\_Banque* d'un attribut de classe : *m\_registreOp* qui modélisera le registre central des numéros d'opérations. La seconde partie de la déclaration sera isolée dans un module source indépendant dédié *\_Banque\_0.cpp*.

Modifier en conséquence la méthode *numeroRegistreOp*.