**Rapport de TP4 IFT3395**

Wen Yin (20179082) Jimmy Yassin Hassanaly (20190749)

***Automne 2023***

**Tâche 3 :**

Voici l’explication des choix des tests pour les tests boîtes blancs des 2 méthodes :

**MainWindow.convert() :**

**testNormalConversion (Couverture des Instructions) :**

Ce test est fait pour exécuter le scénario le plus normal de conversion de devise. Il s'assure que le chemin principal de la méthode convert est exécuté, ce qui inclut le calcul de la conversion de devise. C'est essentiel pour s'assurer que les instructions de base du code fonctionnent comme prévu.

**testUnsupportedCurrencyConversion (Couverture des Arcs) :**

En utilisant des devises inexistantes ("XXX" et "YYY"), ce test vise à parcourir une branche spécifique dans le code si elle existe qui gère les devises non reconnues. Ce test est essentiel pour vérifier que le code se comporte correctement quand il **rencontre des devises qu'il ne peut pas traiter, couvrant ainsi les transitions de contrôle** dans le code pour les entrées inattendues.

**testNegativeAmountConversion (Couverture des Chemins) :**

Ce test vérifie le comportement du code lorsque le montant de conversion est négatif, ce qui devrait normalement être traité comme une entrée invalide. Il s'assure que la méthode « convert» gère correctement ce cas particulier, couvrant un chemin d'exécution distinct qui pourrait impliquer une logique de validation ou de gestion des erreurs.

**testZeroAmountConversion (Couverture des Conditions) :**

Ce test évalue comment le code réagit à une conversion avec un montant de zéro. Cela permet de tester une condition frontière importante (le montant étant exactement zéro), ce qui est important pour s'assurer que les cas limites sont correctement gérés dans le code.

**testAmountAboveLimitConversion (Couverture des i-Chemins) :**

Ce test explore un scénario où le montant est juste au-dessus de la limite autorisée. Il teste les interactions entre les conditions de limite de montant et de validité de la devise, couvrant des cas plus complexes où différentes branches du code interagissent.

**testValidCurrenciesInvalidAmount (Couverture des Conditions) :**

Ce test vise à vérifier le comportement de la méthode convert lorsque les devises fournies sont valides, mais le montant est invalide (dans ce cas, négatif). Cela permet de tester une branche spécifique du code où la validation du montant est effectuée, assurant que le code gère correctement les montants invalides, même quand les devises sont correctes.

**testUnsupportedCurrenciesAndInvalidAmount (Couverture des i-Chemins) :**

En fournissant à la fois des devises non supportées et un montant invalide, ce test évalue comment le code gère la combinaison de plusieurs conditions d'erreur. Cela aide à couvrir des chemins dans le code où différentes branches de logique se croisent, fournissant une vue d'ensemble de la gestion des erreurs plus compliquée.

**testAmountAtUpperLimit (Couverture des Conditions) :**

Ce test vérifie le comportement du code pour un montant qui est exactement à la limite supérieure de la plage acceptée. Cela permet de tester une condition limite spécifique, s'assurant que la méthode convert traite correctement les montants qui sont à la frontière de la validité.