

Laboratoire du 6 octobre

Consignes

Vous trouverez sur le repository GitHub Henallux (<https://github.com/samiroquai/Henalux>) une solution Visual Studio qui constituera votre point de départ pour ce laboratoire.

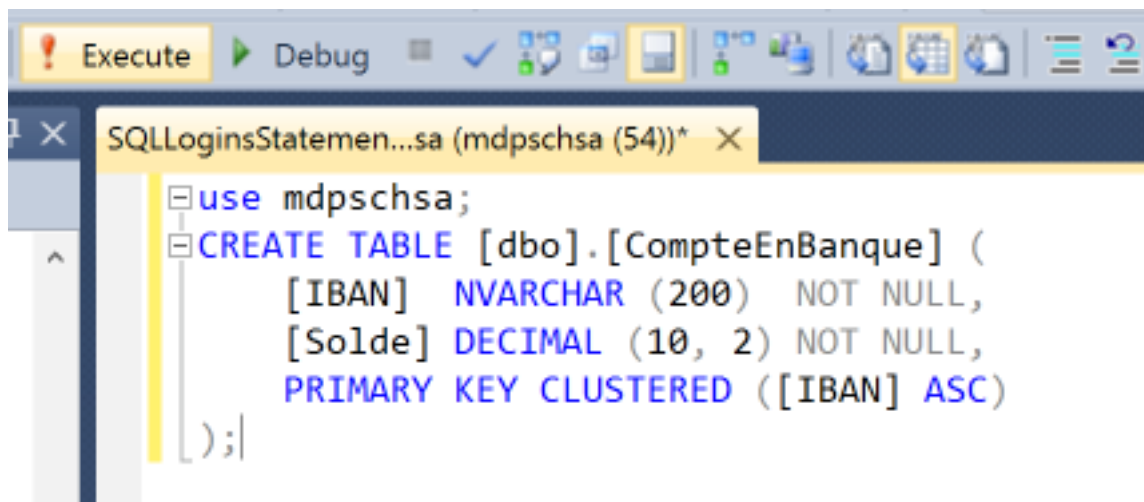
Le domaine d'application est celui d'une banque et plus précisément de la gestion du transfert d'argent entre deux comptes. Un transfert est composé de deux étapes :

- Retrait d'un compte
- Ajout sur un autre compte

Préparation de la base de données

Pour mener à bien ce laboratoire, vous aurez besoin d'une base de données : celle contenant les comptes de la banque. Cette base de données doit être créée sur le serveur vm-sql.iesn.be\Stu3IG mis à disposition par l'IESN (accessible par Internet). Vos credentials vous ont été envoyés par e-mail. Si vous n'avez pas reçu cet e-mail, demandez moi les données. Démarrez une nouvelle instance de SQL Server Management Studio (SSMS) et à l'invite de connexion, utilisez les informations de connexion contenues dans l'e-mail. Pour information, le nom du serveur est constitué de deux éléments : Nom d'hôte\Nom d'instance, à savoir vm-sql.iesn.be\Stu3IG.

Vous n'avez accès qu'à votre base de données (voir mail pour son nom). Depuis SSMS, ouvrez le script SQL contenu dans le repository Git dans l'arborescence suivante : Labo2\BankDAL.Tests\Resources\CreateTable.sql. Dans ce script, insérez le statement « USE XXXXXX ; » où XXXX est à remplacer par le nom de votre base de données. Exécutez ensuite ce script.



Solution Visual Studio

Dans le repository cloné, ouvrez la solution Labo2. Cette solution Visual Studio contient deux projets : un projet contenant la logique de gestion des comptes et un projet de tests de cette logique.

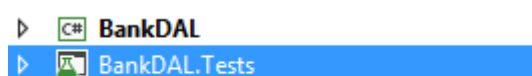


Figure 1 - solution de départ

C'est la classe BankAccountManager qui contient la logique nécessaire aux opérations sur les comptes. Les informations nécessaires sont stockées dans une base de données qui sera recréée à chaque exécution de vos tests.

La classe BankAccountManagerTests (projet de tests) contient quelques tests que vous pouvez lancer. Avant d'aller plus loin dans ce laboratoire, vous devriez arriver dans l'état suivant de l'exécution de vos tests :

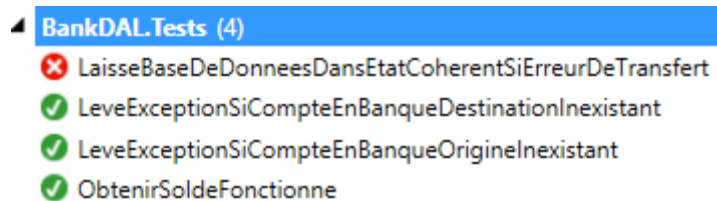


Figure 2 - état attendu des tests

Si tous les tests sont en échec et que les messages d'erreur semblent indiquer un problème de connexion à la base de données, vérifiez les connection string (voir classes BankAccountManager) et plus précisément le nom du serveur SQL et son instance, de même que vos credentials (vos credentials vous ont été envoyés par e-mail, si vous ne les avez pas reçus, demandez-les moi).

Une fois arrivé dans l'état représenté par la Figure 2 vous pouvez poursuivre.

Un des tests ne fonctionne pas. Votre but est de comprendre la raison de cet échec et d'identifier la manière d'y remédier. Vous devez faire passer ce test au vert.

Ne marche pas car il faut ajouter une transaction