



**CÉGEP
SHAWINIGAN**

Du savoir et des gens

Projet final

Projet final (60%)

2022-09-07

Objectif

Mettre en pratique les notions de base de données à l'intérieur d'une application.

Transposer la théorie des requêtes SQL dans une programmation orientée objet.

Comprendre le mécanisme sous-jacent d'un ORM classique en appliquant par le code les fonctions de "**create, drop, select, insert, delete**" et toutes autres fonctions SQL pertinentes à la résolution du problème.

Énoncé

Vous devez développer une application de gestion. Vous avez le choix de l'article géré. Vous devez concevoir votre schéma de base de données en respectant les 3 premières formes normales. Ce programme doit être complet et permettre la gestion de la structure de la base de données, l'affichage des données ainsi que la modification et la suppression.

Besoins non fonctionnels

- L'application doit être développée en C#
- L'application est une application console
- Vous devez utiliser `MySQL.Data.MySqlClient`
- L'utilisation de « Entity » Framework ou autre n'est pas permise
- L'importation d'un script ou fichier SQL n'est pas permise
- Les instructions SQL sont dans le programme
- Vous devez déterminer votre schéma de base de données
- Vous devez avoir minimum 4 tables qui sont partiellement en relation avec des clés étrangères.
 - ***Votre sujet et votre diagramme doivent être approuvés par l'enseignant avant de commencer votre code.***
- Vous devez avoir un dossier "**migrations**" qui contiendra les classes de migration de votre projet.
 - Chaque table doit être représentée par une classe qui permet de la créer, soit une classe de type migration.
 - Par exemple : `itemMigration.cs`, `item2Migration.cs`.
 - Chaque classe pour créer une table doit posséder la fonction « ***migrate, rollback, seed*** » qui contiendra le code pour la gestion de la structure de ladite table
- Vous devez avoir un dossier modèle.
- Vous devez avoir une classe de connexion. Cette classe doit encapsuler et gérer la connexion complète.
 - L'ouvrir, la fermer et donner accès au véritable objet `MySqlConnection` par l'entremise d'une fonction

Besoins fonctionnels

Liste des besoins fonctionnels de votre application

- L'application offre les options de ligne de commande suivantes :
 - --migrate : pour créer les tables.
 - **Celle-ci doit pouvoir être exécutée plusieurs fois et appliquer seulement les nouvelles migrations ajoutées au projet afin d'appliquer des MAJ.**
 - --rollback : supprimer toutes les tables
 - --refresh : recréer la base de données, soit supprimer toutes les tables et les recréer en une seule action. Elle doit réutiliser les fonctions existantes.
 - --seed: insert des données fictives dans la base de données
- On peut combiner certaines options.
 - --migrate --seed: va créer la base de données et ajouter les données fictives
 - --refresh --seed : va supprimer les tables, recréer la base de données et ajouter les données fictives
 - La combinaison doit être fonctionnelle, peu importe l'ordre des paramètres.
- Si l'application n'est pas démarrée avec des paramètres, celle-ci démarre pour afficher la liste des 10 premiers items.
- L'application offre les fonctions suivantes :
 - Suivant : Affiche les 10 items suivants qui n'ont pas été supprimés
 - Précédent : Affiche les 10 items précédents qui n'ont pas été supprimés.
 - Revenir à la page 1 : Affiche les 10 premiers items qui n'ont pas été supprimés.
 - Afficher {id} : Afficher l'item avec l'id en paramètre et son détail
 - Si c'est un item supprimé l'application affiche au début du détail **SUPPRIMÉ**
 - **Supprimer {id} : Supprime avec la technique de SOFT DELETE l'item dont l'id est en paramètre**
 - **Soft delete : Ne supprime pas réellement l'item mais affecte un « flag » qui détermine si l'item est supprimé ou non**
 - **Ajouter : permet d'ajouter un nouvel item en demandant les bonnes informations**

*Un retrait de 10 % peut être retiré pour la qualité du code :

- Indentation uniforme
- Nom de variable représentatif
- Utilisation inappropriée de niveau d'accès privé, protégé et public.
- Nom de fonction représentatif
- Espacement
- Lisibilité générale
- 1 fonction à une seule utilité

Critère de corrections

Les besoins non fonctionnels sont respectés	20 %
Les options de ligne de commande, ' migrate, rollback, refresh et seed ' sont présentes, fonctionnelles et respectes une bonne qualité de code.	10 %
Le programme permet l'affichage des données selon les critères de pagination et les options suivantes, précédent et retour à la page 1	10 %
Le programme implémente correctement le ' soft delete '	5 %
Le programme permet l'ajout d'un nouvel item par l'utilisateur	15 %

Remise :

Date de remise : Vendredi 23 septembre 23 :59

Vous devez remettre :

- Votre diagramme de base de données
- Votre dépôt final du code est dans Bitbucket
- Une remise régulière sur Bitbucket est suggérée tout au long du projet. (Minimum tous les vendredis.)

Références :

- <https://dev.mysql.com/doc/connector-net/en/connector-net-connections-string.html>
- <https://dev.mysql.com/doc/connector-net/en/connector-net-programming-mysqlcommand.html>
- <https://dev.mysql.com/doc/connector-net/en/connector-net-tutorials-sql-command.html>
- <https://docs.microsoft.com/en-us/dotnet/api/system.console?view=net-6.0>
- <https://docs.microsoft.com/en-us/visualstudio/get-started/csharp/tutorial-console?view=vs-2022>