

<p>420-410-SF Bases de données Travail pratique #3 – Partie 3 ORM – Entity Framework</p>

Pondération : 15 % (Total)

Équipe de 2

Mandat général

Cas Tutorat par les pairs au Cégep de Sainte-Foy

- **PARTIE 3**

Implanter un patron de conception Repository pour masquer les détails relationnels de LINQ, Entity Framework (ORM) et C#.

Mandat

- Réaliser une base de données relationnelle simple avec Entity Framework Code-First;
- Peupler les tables de la base de données avec des méthodes de Entity Framework;
- Programmer des requêtes simples d'interrogation sur la base de données avec LINQ Method Syntax.

Livrable C : Implantation du patron de conception Repository, pour cacher les détails d'implantation d'Entity Framework (ORM)

Requis :

- Utiliser le patron de conception Repository (Interface générique, classe générique, utilisation du Repository dans l'application)

Remise :

- À remettre, via Léa (ou Git)
 - o Une mise à jour du projet de type 'Application Console', créé lors de la première partie.

Date de remise : mardi le 19 mai 2015, fin de journée (23 h 55)

SECTION A – Interface générique IRepository

Création de l'interface IRepository (générique)

Dans votre répertoire Repository, créer l'interface générique IRepository.

Les méthodes suivantes devraient apparaître dans le Repository :

- GetAll()
- GetById(int id)
- DeleteAll()
- Delete(Entity)
- Add(Entity)
- Update(Entity)

SECTION B – Implantation de l'interface générique dans une classe générique, BaseRepository

Implantation de l'interface générique dans une classe générique

Attention : Le contexte de base de données doit être passé en paramètre, pour casser la dépendance de la classe BaseRepository

Implanter les méthodes de l'interface IRepository dans la classe BaseRepository. La classe BaseRepository doit rester générique.

SECTION C – Mise à jour de la solution, afin d'utiliser le Repository comme abstraction

Refactorisation du code, pour l'utilisation efficace du Repository

Étape 1 : Créer une classe Application, qui contiendra une instance de la classe BaseRepository pour chaque entité.

- Le constructeur de la classe Application recevra en paramètres un BaseRepository
- Toutes les méthodes d'interrogation présentement dans votre Main, seront transférées à votre classe Application.
- Dans ces méthodes d'interrogation, où le contexte de base de données est utilisé directement, substituer maintenant le Repository afin de faire abstraction complètement de la base de données et des ORMs.

Étape 2 : Une fois que l'abstraction sera faite, dans votre Main, vous n'aurez qu'à instancier votre classe Application et votre Repository. Utiliser ensuite les méthodes de la classe Application pour afficher les mêmes données qu'avant (les requêtes du TP2).

