

TP 4 – Accès aux données

Objectif du TP :

Gérer la connexion à une base de données SQL Server, échanger des objets entre la base et l'application récupérée du TP précédent. Tester la couche d'accès aux données avec des tests unitaires.

Etape 1 : Une nouvelle couche

Le but de ce TP est de gérer l'accès à votre tournoi depuis une base de données en lieu et place de l'initialisation en dur actuelle. C'est-à-dire remplacer l'accès aux données « Stubé » (ou bouchonné) par votre nouvelle couche Data Access Layer.

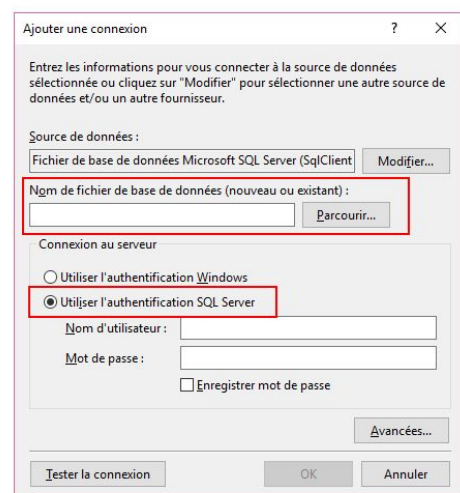
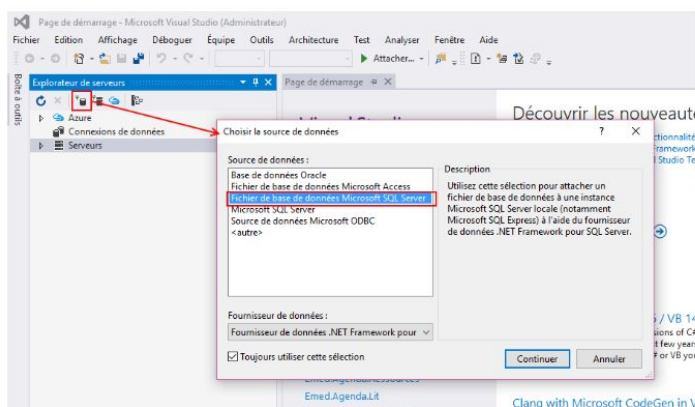
Ajouter un nouveau projet « Bibliothèque de classes » à votre application. Cette nouvelle librairie Data Access Layer ne sera en référence que de la couche Business (la visibilité entre couche se fait de proche en proche) et de la couche Entities.

Pour la couche Présentation, la mise à jour est transparente. Au niveau de la couche Business, l'impact est localisé au niveau du manager.

Etape 2 : L'accès aux données

Pour les besoins du TP vous devez créer une base de données. Vos professeurs préférés vous laissent le choix pour cette dernière : vous pouvez créer votre base de données sur Windows Azure ou un un fichier de base de données SQL voir même MySQL.

Créer un fichier de base de données SQL Server



Implémenter l'accès aux données via un design pattern Bridge comme vu en cours.

La Dal met à disposition de la couche Business des List<Pokemon>, ainsi que les méthodes nécessaires pour :

- Récupérer les matchs sous la forme d'une List<Match>
- Récupérer les stades sous la forme d'une List<Stade>
- Récupérer les caractéristiques sous la forme d'une List<Caracteristique>
- Créer ensuite les méthodes nécessaires pour ajouter/supprimer/mettre à jour les différents objet de la base de données.

La couche Business ne voit de la Dal que son manager singletonisé.

Etape 4 : Ajout des tests unitaires

Créer un nouveau projet de Tests. Ce projet de tests contient une référence à la couche Dal. Son but est de pouvoir la tester. Pour cela il contient donc l'ensemble des méthodes de tests CRUD. Chacune de ses méthodes accèdent au singleton d'accès aux données afin de tester séparément l'accès, la création, la modification ainsi que la suppression d'un Pokemon.

Conclusion

A la fin de ce TP, vous avez une smart application WPF de gestion de votre tournoi qui permet l'alimentation d'une base de référence. Ce qui termine la partie Smart Application du cours.