

## LINQ & Entity Framework – Exercices

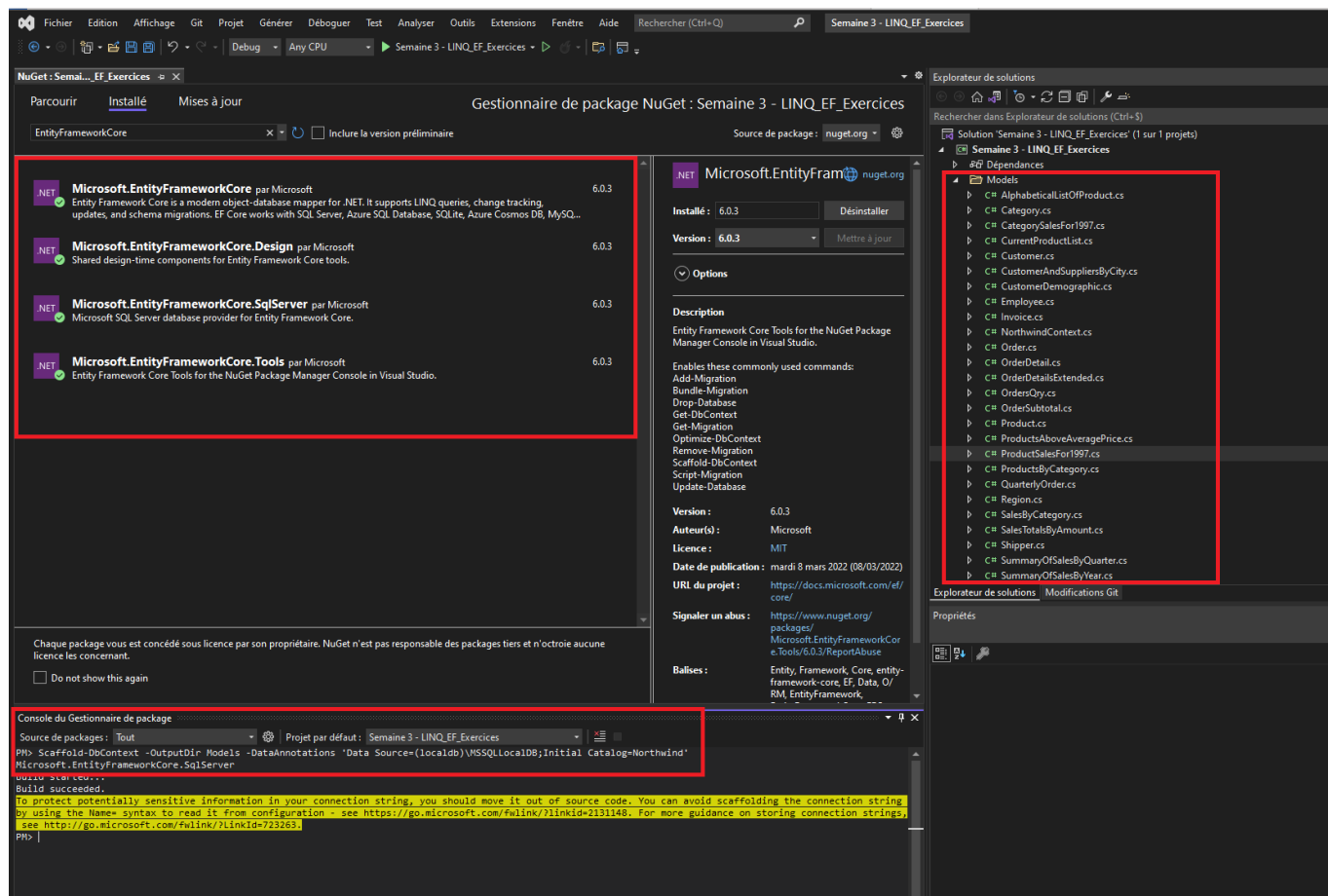
### Objectifs

- ✓ Créer un modèle à partir d'une base de données existante
- ✓ Manipuler la base de données au travers de LINQ (ajout, suppression, recherche)
- ✓ Utiliser les propriétés de navigation

### A – Création des entités (DAO) à partir de la base de données

1. Créer un projet Application Console
2. Créer la base de données Northwind à partir du fichier « Northwind4SqlServer.sql »
  - a. Récupérer sur MooVin le fichier « Northwind4SqlServer.sql » dans les ressources
  - b. Exécuter un query avec ce fichier SQL dans Visual Studio
    - i. Outils → SQL SERVER → Nouvelle requête/query
    - ii. Copier-coller le code du fichier « Northwind4SqlServer.sql »
    - iii. Exécuter le code du query (bouton play vert en haut à gauche de la fenêtre query)
3. Voir la base de données Northwind
  - a. Dans Visual Studio, vous pouvez manipuler les bases de données SQL SERVER via l'explorateur d'objets SQL SERVER
  - b. Affichage → Explorateur d'objets SQL SERVER
  - c. Vous devez vous connecter à une instance SQL Server. Nous utiliserons toujours l'instance : **(localdb)\MSSQLLocaldb**
  - d. Vérifier que vous avez bien une base de données Northwind et afficher les données d'une table
4. Ajouter les packages NuGet nécessaires pour EntityFramework
  - a. Clic droit sur le projet → Gérer les packages NuGet
  - b. Ajouter les packages
    - i. Microsoft.EntityFrameworkCore
    - ii. Microsoft.EntityFrameworkCore.Design
    - iii. Microsoft.EntityFrameworkCore.Tools
    - iv. Microsoft.EntityFrameworkCore.SqlServer
5. Créer les classes-entités depuis la DB
  - a. **Dans une console du Package Manager (PM)**
    - i. Outils – Gestionnaire de Package NuGet → Console
  - b. Lancez la commande (sur une seule ligne) :
6. **Scaffold-DbContext -OutputDir Models 'Data Source=(localdb)\MSSQLLocalDB;Initial Catalog=Northwind' Microsoft.EntityFrameworkCore.SqlServer**

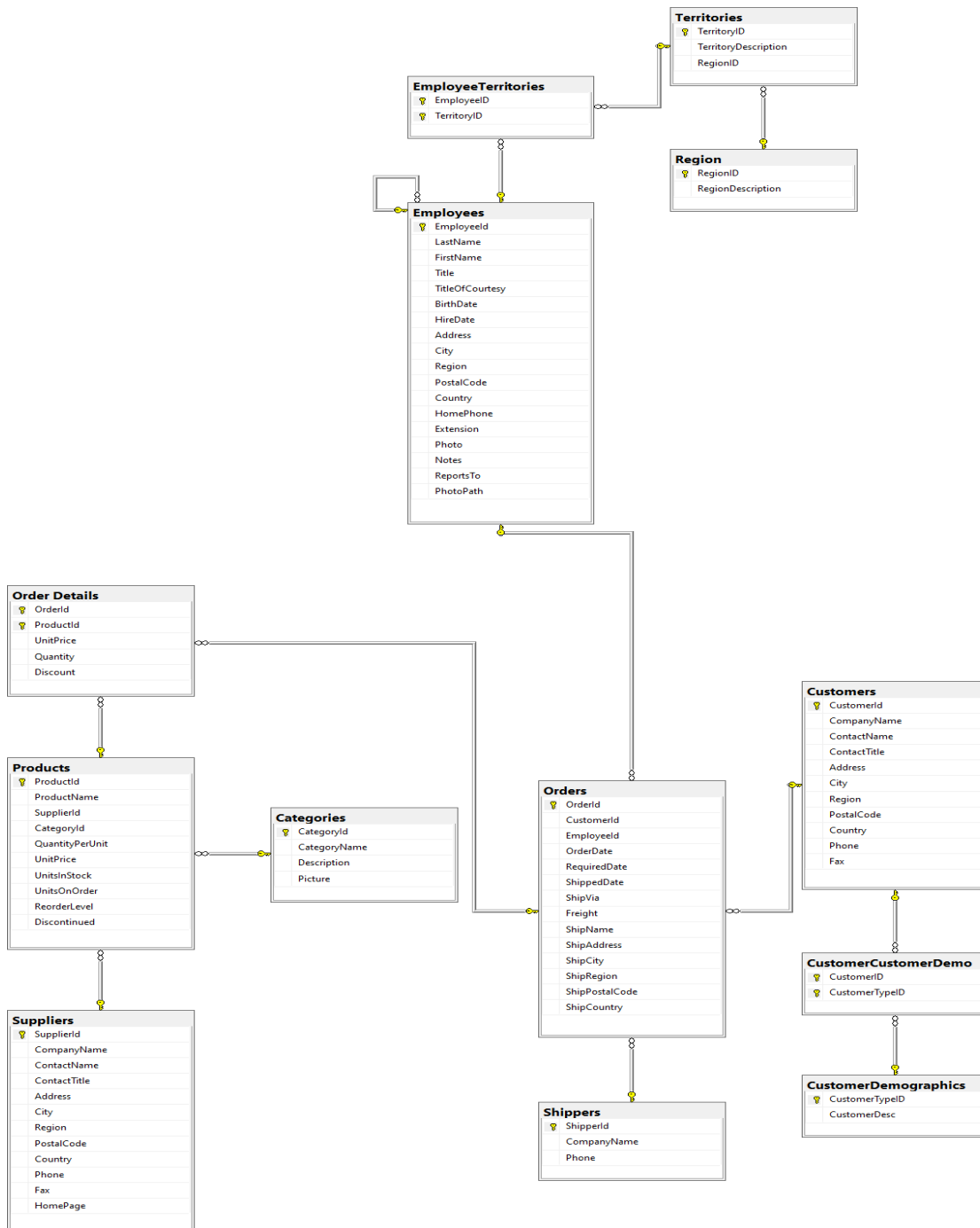
En image :



7. Regardez ce qui a été généré par la commande dans le dossier Models
  - a. Regardez également les fichier C# générés
    - i. Le fichier étendant DbContext (point d'entrée pour manipuler la DB)
    - ii. Les POCOS créés -> les entités correspondant aux tables
    - iii. Pq les classes sont-elles partielles ?
  - iv. Pq les propriétés de navigation sont-elles virtual ?

## B – Queries

Un petit schéma de la DB :



Afin de pouvoir manipuler la base de données via l'Entity Framework, il vous faudra dans votre *Main()* créer un objet du type *NorthwindContext* (la classe générée étendant *DbContext*). Le nom de cette classe peut être différent si vous avez changé le nom par défaut dans l'assistant. Essayez maintenant de réaliser les requêtes LINQ suivantes :

0. Activer le lazy loading avec utilisation d'un proxy
  - a. Installer le package *EntityFrameworkCore.Proxies*
  - b. Aller dans votre fichier *DbContext* *Northwind* généré lors de l'étape A.
  - c. Modifier la méthode *OnConfiguring* comme ceci :

```
protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)
```

```

{
    if (!optionsBuilder.IsConfigured)
    {
        optionsBuilder.UseSqlServer("Data Source=(localdb)\\MSSQLLocalDB;Initial
        Catalog=Northwind;MultipleActiveResultSets=True")
            .UseLazyLoadingProxies()
            .LogTo(Console.WriteLine, LogLevel.Information)
            .EnableSensitiveDataLogging();
    }
}

```

1. Lister tous les *Customers* habitants dans une ville saisie au clavier.

```

B1 - Recherche clients par ville
Entrez le nom d'une ville :
Bruxelles
MAISD : CATHERINE DEWEY

```

2. Afficher les produits de la catégorie Beverages et Condiments. Utilisez **le lazy loading** (pas d'include !)

Résultat attendu :

```

Catégorie : Beverages
Chai
Chang
Guaraná Fantástica
Sasquatch Ale
Steeleye Stout
Côte de Blaye
Chartreuse verte
Ipoh Coffee
Laughing Lumberjack Lager
Outback Lager
Rhönbräu Klosterbier
Lakkalikööri
Catégorie : Condiments
Aniseed Syrup
Chef Anton's Cajun Seasoning
Chef Anton's Gumbo Mix
Grandma's Boysenberry Spread
Northwoods Cranberry Sauce
Genen Shouyu
Gula Malacca
Sirop d'érable
Vegie-spread
Louisiana Fiery Hot Pepper Sauce
Louisiana Hot Spiced Okra
Original Frankfurter grüne Soße

```

3. Afficher les produits de la catégorie Beverages et Condiments. Utilisez **le eager loading** ! (avec include). Le résultat est identique à la requête précédente.
  - a. Désactiver le lazy loading -> retirer `.UseLazyLoadingProxies()`
  - b. Remarquez que vous avez maintenant un seul select !

4. Donnez pour un client donné saisi au clavier (LILAS par ex) la liste de ces commandes (de la plus récente à la plus ancienne) et qui ont été livrées ( il y a une date de livraison). Les champs renvoyés par ce query sont le ID du client «CustomerID », la date de la commande « OrderDate » et la date de livraison « ShippedDate ».

```
Entrez l'ID d'un client
LILAS
CustomerID : LILAS OrderDate : 03/04/1998 00:00:00 ShippedDate :13/04/1998 00:00:00
CustomerID : LILAS OrderDate : 20/02/1998 00:00:00 ShippedDate :26/02/1998 00:00:00
CustomerID : LILAS OrderDate : 09/01/1998 00:00:00 ShippedDate :13/01/1998 00:00:00
CustomerID : LILAS OrderDate : 16/12/1997 00:00:00 ShippedDate :25/12/1997 00:00:00
CustomerID : LILAS OrderDate : 21/05/1997 00:00:00 ShippedDate :23/05/1997 00:00:00
CustomerID : LILAS OrderDate : 08/04/1997 00:00:00 ShippedDate :16/04/1997 00:00:00
CustomerID : LILAS OrderDate : 28/02/1997 00:00:00 ShippedDate :05/03/1997 00:00:00
CustomerID : LILAS OrderDate : 12/12/1996 00:00:00 ShippedDate :13/12/1996 00:00:00
CustomerID : LILAS OrderDate : 19/11/1996 00:00:00 ShippedDate :02/12/1996 00:00:00
CustomerID : LILAS OrderDate : 16/10/1996 00:00:00 ShippedDate :28/10/1996 00:00:00
CustomerID : LILAS OrderDate : 03/09/1996 00:00:00 ShippedDate :11/09/1996 00:00:00
CustomerID : LILAS OrderDate : 16/08/1996 00:00:00 ShippedDate :23/08/1996 00:00:00
```

5. Afficher le total des ventes par produit (ID produit -> Total) trié par ordre de numéro produit.

```
1 ----> 14277,6000
2 ----> 18559,2000
3 ----> 3080,0000
4 ----> 9424,8000
5 ----> 5801,1500
6 ----> 7345,0000
7 ----> 22464,0000
8 ----> 13760,0000
9 ----> 8827,0000
10 ----> 22140,2000
11 ----> 13902,0000
12 ----> 12866,8000
13 ----> 5234,4000
14 ----> 8630,4000
15 ----> 1813,5000
16 ----> 18748,0500
17 ----> 35482,2000
18 ----> 31987,5000
19 ----> 6159,5000
20 ----> 23635,8000
21 ----> 9636,0000
22 ----> 7232,4000
23 ----> 4840,2000
24 ----> 4782,6000
25 ----> 4051,6000
26 ----> 21534,9000
27 ----> 15231,5000
28 ----> 26865,6000
29 ----> 87736,4000
30 ----> 14775,5400
```

.....

6. Afficher tous les employés (leur nom) qui ont sous leur responsabilité la région « Western »

```
Liste des employés de la région Western
Suyama Michael
King Robert
```

7. Quels sont les territoires gérés par le supérieur de « Suyama Michael »

```
Les territoires gérés par le supérieur de Suyama  
Providence  
Morristown  
Edison  
New York  
New York  
Mellville  
Fairport
```

## C – Updates

1. Mettez en majuscule le nom de tous les clients
2. Vérifiez que l'update a bien été réalisé en faisant un query qui affiche les noms des clients

## D – Insert

1. Ajoutez une catégorie à partir d'un nom saisi au clavier
2. Vérifiez que l'ajout a bien été effectué en DB

## E – Deletes

1. Supprimez la catégorie ajoutée à l'exercice précédent
2. Vérifiez que la suppression a bien été faite en DB
3. Supprimez un employé et réassignez toutes ses commandes (« orders ») à un autre employé.  
Vous demanderez les «id » des employés au clavier. Essayez de supprimer DAVOLIO  
(employeeId ==1) et de réassigner ses commandes à FULLER (employeeId ==2)
4. Vérifiez