

# Entity Framework Core

---

# Mémos pour Entity Framework Core

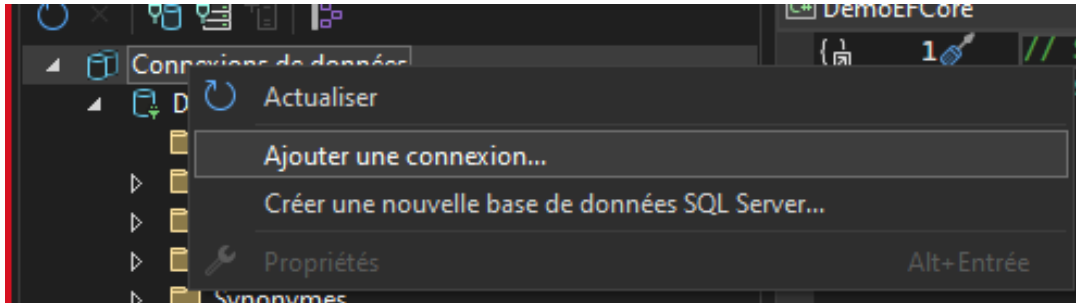
## Création d'une BDD locale SQL Server

Pour créer une Base de données SQL Server locale sur Windows, utiliser ces commandes dans un terminal :

```
PS C:\Users\Administrateur> sqllocaldb c DemoEfcore  
LocalDB instance "DemoEfcore" created with version 15.0.4153.1.  
PS C:\Users\Administrateur> sqllocaldb s DemoEfcore  
LocalDB instance "DemoEfcore" started.
```

# Connexion à une BDD locale SQL Server via visual Studio

Dans l'explorateur de serveur :



Ajouter une connexion ? X

Entrez les informations pour vous connecter à la source de données sélectionnée ou cliquez sur "Modifier" pour sélectionner une autre source de données et/ou un autre fournisseur.

Source de données :  
Microsoft SQL Server (SqlClient) Modifier...

Nom du serveur :  
(LocalDB)\DemoEFCore Actualiser

Connexion au serveur

Authentification : Authentification Windows

Nom d'utilisateur :  
Mot de passe :  
☐ Enregistrer mon mot de passe

Connexion à la base de données

☒ Sélectionner ou entrer un nom de base de données :  
[Sélectionner]

☐ Attacher un fichier de base de données :  
[Parcourir...]  
Nom logique :  
[Saisir]

Avancées...

Tester la connexion OK Annuler

*Nb : Penser à renommer la base pour la différencier facilement ensuite*

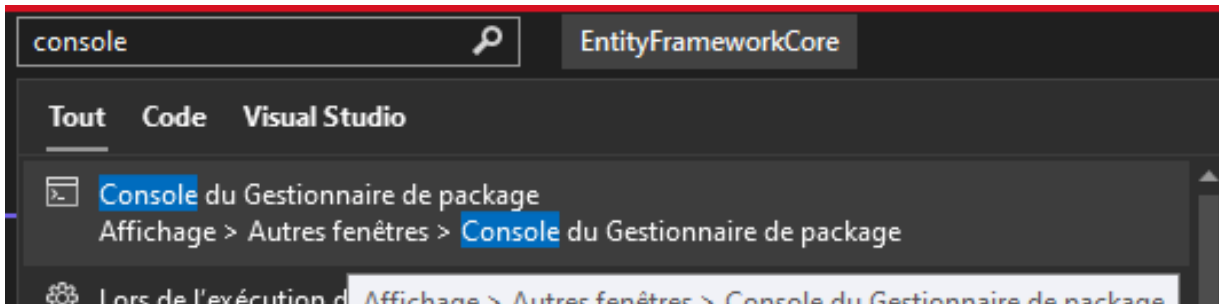
# Packages NuGet nécessaires à l'utilisation d'EFCore

.NET ↑	<b>Microsoft.EntityFrameworkCore</b> par Microsoft Entity Framework Core is a modern object-database mapper for .NET. It supports LINQ queries, change tracking, updates, and schema migrations. EF Core works with SQL Server, Azure SQL Database, SQLite, Azure Cosmos DB, MySQL, PostgreSQL, and other databases through a provider plugin API.	6.0.16
		7.0.5
.NET ↑	<b>Microsoft.EntityFrameworkCore.SqlServer</b> par Microsoft Microsoft SQL Server database provider for Entity Framework Core.	6.0.16
		7.0.5
.NET ↑	<b>Microsoft.EntityFrameworkCore.Tools</b> par Microsoft Entity Framework Core Tools for the NuGet Package Manager Console in Visual Studio.	6.0.16
		7.0.5

*La dernière version 6.X.X est la plus compatible avec le DotNet6*

# Ajout de Migrations avec les Tools et le Package Manager

Accéder à la console du gestionnaire de package



Taper cette commande :

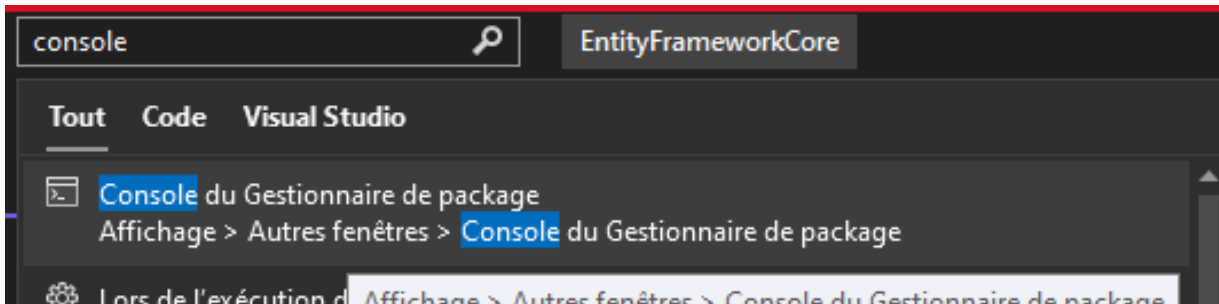
```
PM> Add-Migration PremiereMigration
Build started...
Build succeeded.
To undo this action, use Remove-Migration.
```

Le projet doit compiler pour pouvoir créer une compilation

La migration est ensuite prête à être appliquée à la BDD

# Mise à jour de la BDD à partir de Migrations

Accéder à la console du gestionnaire de package



Taper cette commande :

```
PM> Update-Database
Build started...
Build succeeded.
Applying migration '20230502081646_PremiereMigration'.
Done.
```

## Commandes utiles du PM

- **Remove-Migration**

Supprime la dernière migration

/!\ Si elle n'est pas appliquée à la BDD

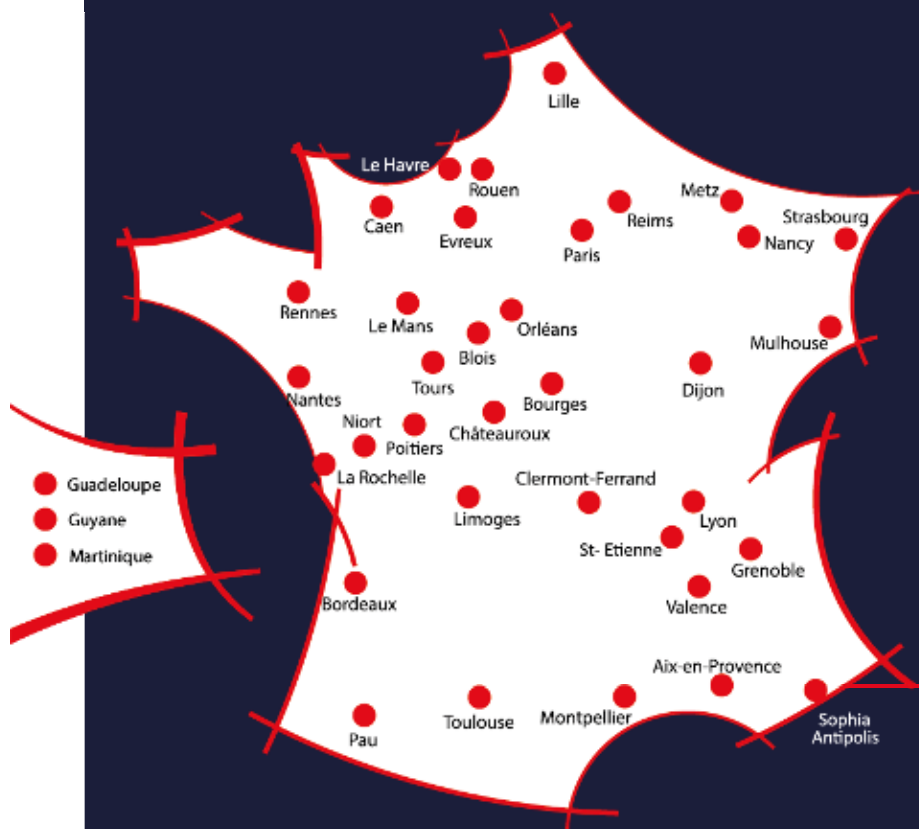
/!\ Ne pas supprimer directement la classe, sinon impossible de l'inverser si elle a été appliquée à la BDD

- **Update-Database 0 / <NomMigration>**

Met à jour la base de donnée à une migration précédente ou à l'état initial (**0**)

/!\ Peut mener à une perte de données





Découvrez également  
l'ensemble des stages à votre disposition  
sur notre site [m2information.fr](https://m2information.fr)

[m2information.fr](https://m2information.fr)

