

TP N° 5

Application web MVC avec Entity Framework Core (Approche Database First)

Partie 1 : Créer la Base de Données dans SQL Server :

Ouvrir SQL Server Management Studio (SSMS) :

(Pour installation : <https://learn.microsoft.com/en-us/ssms/install/install>)

1.1. Créer une base de données, ex. : BD_VENTE :

```
CREATE DATABASE BD_VENTE;  
GO
```

1.2. Utiliser la base :

```
USE BD_VENTE;  
GO
```

1.3. Créer une table PRODUITS :

```
CREATE TABLE PRODUITS (  
    ID INT PRIMARY KEY IDENTITY(1,1),  
    NOM VARCHAR(100) NOT NULL,  
    DESCRIPTION VARCHAR(255),  
    PRIX DECIMAL(10,2) NOT NULL,  
    QUANTITE INT NOT NULL,  
    DATE_AJOUT DATE DEFAULT GETDATE(),  
    DISPONIBLE BIT DEFAULT 1  
);
```

1.4. Insérer des données, exemple :

```
INSERT INTO PRODUITS (NOM, DESCRIPTION, PRIX, QUANTITE)  
VALUES  
('Ordinateur Portable', 'PC 15 pouces, i5, 8Go RAM', 799.99, 12),  
('Souris Optique', 'Souris USB avec capteur optique', 19.99, 50);
```

Partie 2 : Créer le projet ASP.NET Core MVC

- Ouvrir Visual Studio
- Aller dans Fichier > Nouveau > Projet
- Choisir : ASP.NET Core Web App (Model-View-Controller)
- Cliquer sur Suivant
- Nom du projet : **App_ACHAT**
- Cliquer sur Créer
- Choisir le Framework .NET 6 ou supérieur
- Laisser les options par défaut, puis cliquer sur Créer

Partie 3 : Installer les packages NuGet pour EF Core

Méthode 1 – Interface graphique : Outils > Gestionnaire de packages NuGet > Gérer les packages NuGet pour la solution

Chercher et installer ces packages :

- Microsoft.EntityFrameworkCore
- Microsoft.EntityFrameworkCore.SqlServer
- Microsoft.EntityFrameworkCore.Tools

Méthode 2 – Console : Outils > Gestionnaire de packages NuGet > Console

- Install-Package Microsoft.EntityFrameworkCore
- Install-Package Microsoft.EntityFrameworkCore.SqlServer
- Install-Package Microsoft.EntityFrameworkCore.Tools

A éviter :

- ⇒ Ne pas oublier d'installer EFCore.Tools, sinon Scaffold-DbContext ne fonctionnera pas.
- ⇒ Ne jamais modifier directement DbContext.cs généré (on utilisera partial ou configuration séparée si besoin).

Partie 4 : Générer les modèles depuis la base de données

1.5. Commande Database First, dans la Console du Gestionnaire de Package :

```
Scaffold-DbContext "Data Source=(localdb)\MSSQLLocalDB;Initial
Catalog=BD_VENTE;Integrated Security=True;TrustServerCertificate=True"
Microsoft.EntityFrameworkCore.SqlServer -OutputDir Models -Context
VenteContext
```

⇒ Résultat :

- Un fichier VenteContext.cs (DbContext)
- Un fichier Produit.cs (modèle)

Partie 5 : Configurer la connexion dans appsettings.json

- Ajouter la partie suivante dans le fichier json existant :

```
"ConnectionStrings": {
  "VenteDb": "Data Source=(localdb)\\MSSQLLocalDB;Initial
Catalog=BD_VENTE;Integrated
Security=True;TrustServerCertificate=True"
}
```

Partie 6 : Ajouter le DbContext dans Program.cs

Dans Program.cs, **avant** var app = builder.Build();

```
builder.Services.AddDbContext<VenteContext>(options =>
options.UseSqlServer(builder.Configuration.GetConnectionString("VenteDb")));
```

Attention :

- ⇒ Cette ligne doit être avant builder.Build();

⇒ S'assurer que `using Microsoft.EntityFrameworkCore;` est présent

Partie 7 : Créer un contrôleur avec vues (scaffolding)


Étapes :

1. Clic droit sur Controllers → Ajouter → Contrôleur
2. Choisir : **MVC Controller with views, using Entity Framework**
3. Sélectionner :
 - **Classe modèle** : Produit
 - **DbContext** : VenteContext
 - Nom du contrôleur : ProduitsController
4. Valider → Visual Studio génère :
 - ProduitsController.cs
 - Les vues Razor (Index, Create, Edit, Details, Delete)

Partie 8 : Exécuter l'application

- Lancer l'application
- Naviguer vers : `https://localhost:xxxx/Produits`
- Visualiser la liste des produits (avec CRUD complet)

Exemple de résultat avec CRUD :



Nom	Description	Prix	Quantite	DateAjout	Disponible	
Ordinateur Portable	PC 15 pouces, i5, 8Go RAM	799,99	12	02/11/2025	<input checked="" type="checkbox"/>	Edit Details Delete
Souris Optique	Souris USB avec capteur optique	19,99	50	02/11/2025	<input checked="" type="checkbox"/>	Edit Details Delete