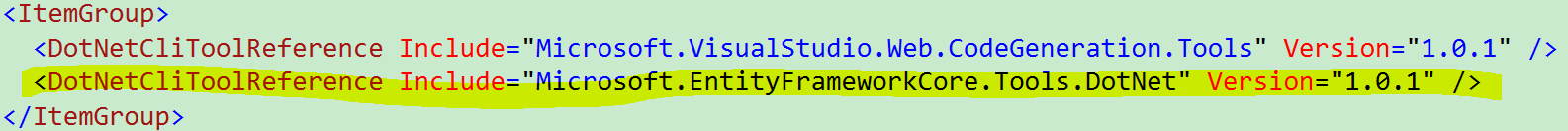
1. how to do migrations
2. add Command-line interface(CLI) tool into project

edit project.csproj file (here is NetCoreApplication.csproj) by right-clicking the project name in "Solution Explorer" and selecting "Edit project.csproj". then add



1. modify appsettings.json file (for getting new DB connection string)

drop old DB by the command

dotnet ef database drop

1. create DB by migrations

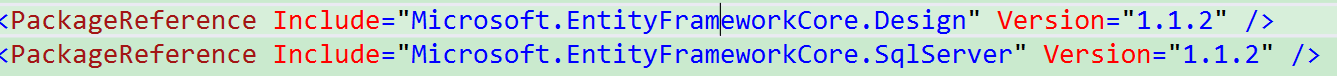
dotnet ef migrations add InitialCreate

after running this command, it will be generate a "Mirgations" folder which includes two files:

one is "Date\_InitailCreate.cs" which include two methods "UP" (create or update DB),"Down" (remove DB or updating)

another is Snapshot file which is the DB schema

Note: if there are error “Could not load file or assembly 'Microsoft.EntityFrameworkCore.Design, Culture=neutral, PublicKeyToken=null'. The system cannot find the file specified.” When running “dotnet ef migrations add InitailCreate”, we can firstly run command “dotnet add package Microsoft.EntityFrameworkCore.Design” and set project as start project.



1. update DB

"dotnet ef database update"

1. remove previous update

"dotnet ef migrations remove"

At the same time, we also can use Package Manager Console to do migrations, the different is package needed to install “ [Microsoft.EntityFrameworkCore.Tools](https://www.nuget.org/packages/Microsoft.EntityFrameworkCore.Tools) ”, refer to https://docs.microsoft.com/en-us/ef/core/miscellaneous/cli/powershell

1. Install package

In package Manager console, execute

Install-Package Microsoft.EntityFrameworkCore.Tools

1. Create DB, add a new migrations

Add-Migration InitialCreate

1. Remove the last migration

Remove-Migration

1. Update DB

Update-Database

1. Drop DB

Drop-Database

1. Generate a sql script

Scrit-Migration

1. Repository and Unit of work pattern

Add generic repository and unit of work into data access layer. Refer to: <https://docs.microsoft.com/en-us/aspnet/core/data/ef-mvc/advanced>