**School Management System MVC Project**

(Database-first Approach)

Design Pattern – IoC(Inversion of Controll)

Docker Configuration:

* SQL Server Image:

docker pull mcr.microsoft.com/mssql/server

* Create Docker Container:

docker run -e 'ACCEPT\_EULA=Y' -e 'SA\_PASSWORD=UPPERlower123@#' -p 1400:1433 -d mcr.microsoft.com/mssql/server

Database:

* Create DB

CREATE DATABASE SchoolDB

GO

USE SchoolDB;

* Create Tables

CREATE TABLE Students(

    Id INT PRIMARY KEY IDENTITY,

    FirstName NVARCHAR(50) NOT NULL,

    LastName NVARCHAR(50) NOT NULL,

    DateOfBirth DATE,

)

CREATE TABLE Lecturers(

    Id INT PRIMARY KEY IDENTITY,

    FirstName NVARCHAR(50) NOT NULL,

    LastName NVARCHAR(50) NOT NULL,

)

CREATE TABLE Courses(

    Id INT PRIMARY KEY IDENTITY,

    Name NVARCHAR(50) NOT NULL,

    Code NVARCHAR(5) UNIQUE,

    Credits INT,

)

* Database Connection String (appsettings.json)

"ConnectionStrings": {

    "DefaultConnection": "Server=localhost, 1400;Database=SchoolDB;Trusted\_Connection=false;MultipleActiveResultSets=true;Encrypt=false;User Id=sa;Password=UPPERlower123@#;"

}

* Database Scaffolding (Generate Code from Database Tables and Fields):

dotnet ef dbcontext scaffold "Server=localhost, 1400;Database=SchoolDB;Trusted\_Connection=false;MultipleActiveResultSets=true;Encrypt=false;User Id=sa;Password=UPPERlower123@#;" Microsoft.EntityFrameworkCore.SqlServer -o Data -f --no-onconfiguring

* Configur eDependancy Injection DBContext in Program.cs

var config = builder.Configuration.GetConnectionString("DefaultConnection");

builder.Services.AddDbContext<SchoolDbContext>(options =>

    options.UseSqlServer(config));

Packages

Installed Packages:

* 1. Microsoft.EntityFrameworkCore
  2. Microsoft.EntityFrameworkCore.Design
  3. Microsoft.EntityFrameworkCore.SqlServer
  4. Microsoft.EntityFrameworkCore.Tools
  5. Microsoft.VisualStudio.Web.CodeGeneration.Design

Terminal CLI Commands:

dotnet add package Microsoft.EntityFrameworkCore

dotnet add package Microsoft.EntityFrameworkCore.SqlServer

dotnet add package Microsoft.EntityFrameworkCore.Design

dotnet add package Microsoft.EntityFrameworkCore.Tools

dotnet add package Microsoft.VisualStudio.Web.CodeGeneration.Design

Tools

Installed Tools:

1. Entity Framework
2. Code Generator

Commands:

dotnet tool install --global dotnet-ef

dotnet tool install --global dotnet-aspnet-codegenerator

Controllers

1. CoursesController (via Codegenerator)

Code Generator CLI Command:

dotnet aspnet-codegenerator controller -name CoursesController -m Course -dc SchoolDbContext --relativeFolderPath Controllers --useDefaultLayout --referenceScriptLibraries -f