EF Core kulllanabilmek için önce EF Core’yi projeye kurmalıyız. Bunun için [EF Core Providers](https://learn.microsoft.com/en-us/ef/core/providers/?tabs=dotnet-core-cli) sitesine giderek .net cli kodunu alıyoruz ve terminale yapıştırıp enterliyoruz.

Migration oluştururken alınan hata:



Bu hata nasıl çözülür?

DataContext classına

public DataContext(DbContextOptions<DataContext> options): base(options)

{

}

Kurucu metodu eklenerek hata giderilir.

Migrationı ekledikten sonra, veri tabanını yeniden düzenlemek istiyorsak var olan migration klasörünü silip yeni migration oluşturabiliriz. Ya da yeni sütunlar eklemek için var olan migrationı silmeden yeni migration oluşturabiliriz.

Migration Oluşturma/64

Migration oluşturmak için uygulamada, program cs dosyasında bir buildera add db context metodu işlettik.

using EntityFrameworkCore.Data;

using Microsoft.EntityFrameworkCore;

var builder = WebApplication.CreateBuilder(args);

// Add services to the container.

builder.Services.AddControllersWithViews();

// bu kod, mutlaka var app = builder.Build(); kodundan önce olmalı

builder.Services.AddDbContext<DataContext>(options => {

var config = builder.Configuration;

var connectionString = config.GetConnectionString("database");

options.UseSqlServer(connectionString);

});

var app = builder.Build();

DataContext dosyamızda, bir kurucu metot ekledik.

public DataContext(DbContextOptions<DataContext> options): base(options)

{

}

Ayrıca appsettings.Development.json dosyamıza connection stringi ekledik.

{

"Logging": {

"LogLevel": {

"Default": "Information",

"Microsoft.AspNetCore": "Warning"

}

},

"ConnectionStrings": {

"database": "Data Source=(localdb)\\MSSQLLocalDB;Initial Catalog=efcore;Integrated Security=True"

}

}