API

SERVICE , EndPointليه؟ عشان انا عايز اشيل الفيو وعايزة اكتب لوجيك بس>> ده اسمه

+بفصل الباك عن الفرنت

Multiple Pages>> ان كل اما افتح بيدج اعمل ريكويست جديد وبنقل من صفحه للتانيه mvc,razor فكرتها ان في

وكل ريجويست بيجيب html بتاع الصفحة

Web API>>عباره عن سيرفيس بديها للفرنت , بدل الاكشن endPointابدا

Consumed in Mobile , Fluter , Android , Ios,

Frontend Framework [React Or Angular]

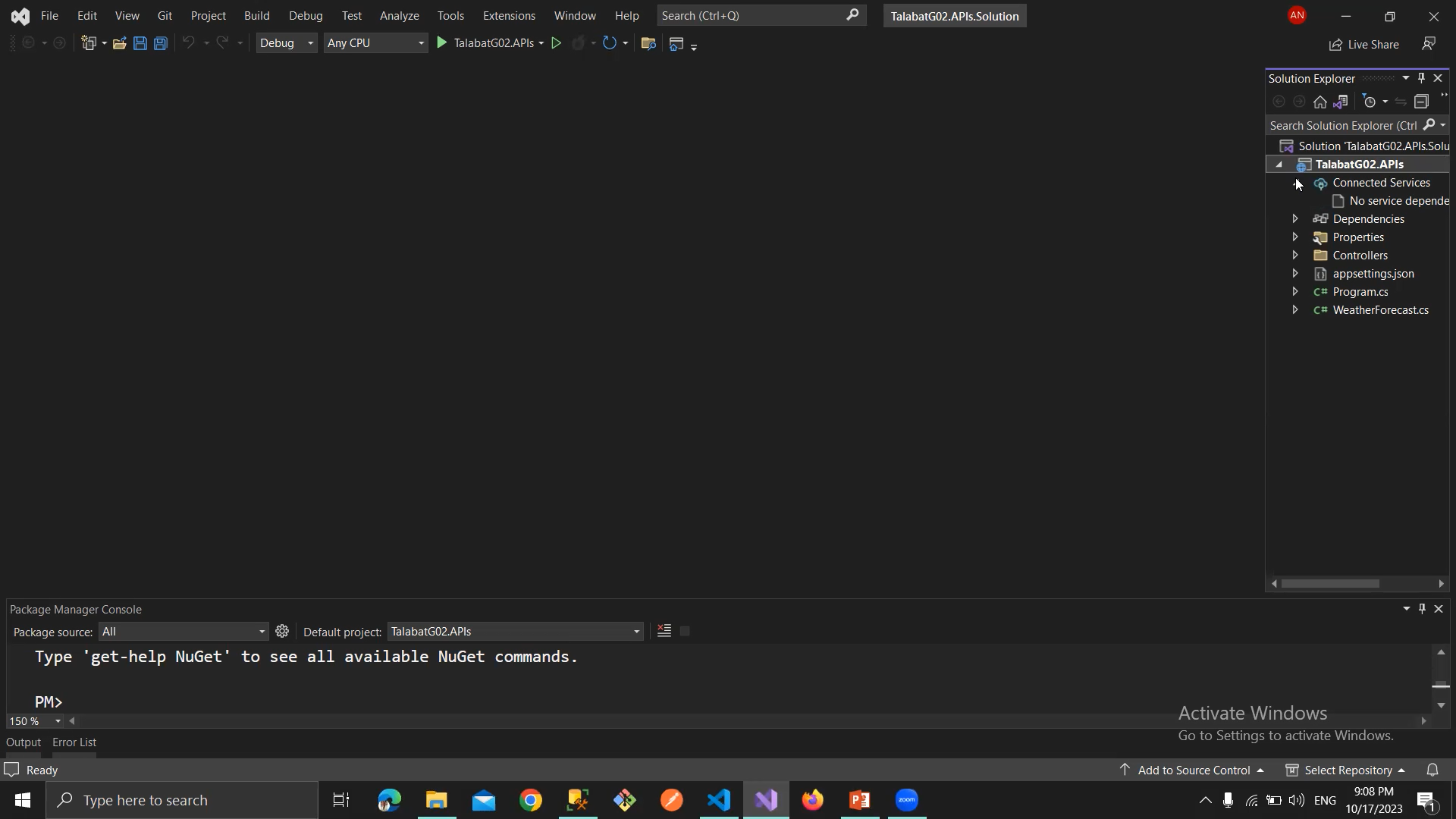
webApi >>Model-Controller(MC)الكونترولر مش بيرجعلى فيو بيرجعلي داتا وبتاع الفرنت اند ياخدها ويعرضها على حسب الفيو بتاعه

(array of obj(بيبقى الداتا جواها عباره عن JSON,XML->الداتا بترجع على هيئه

**MVC>>** **Controller => Contains Functions Named As Actions -> Return View**

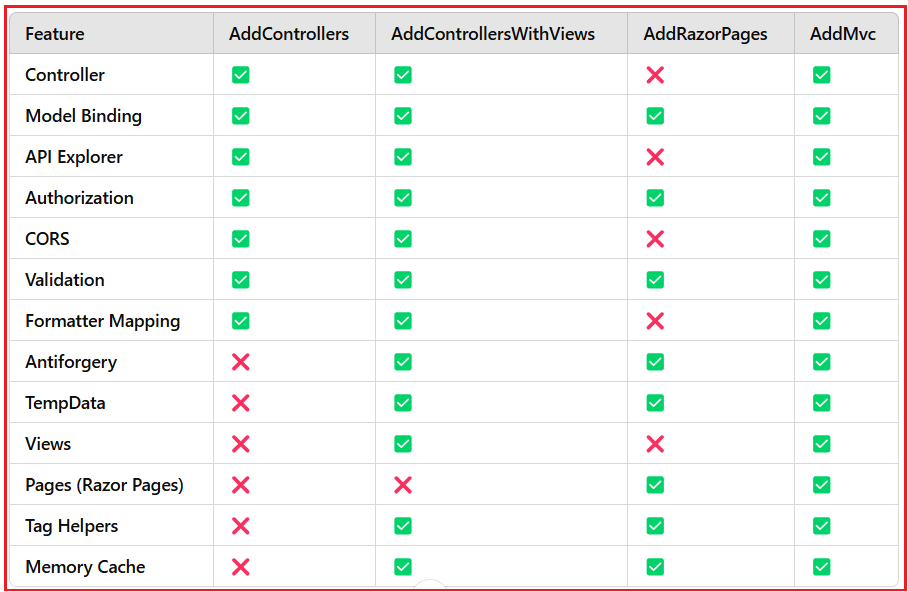
**API>>** **Controller => Contains Functions Named As EndPoints -> Return JSON Data**

MVCامتى استخدم ده وامتى استخدم ده >لو الريبوزاتري فيها قليله اواليوزر قليلين نستخدم



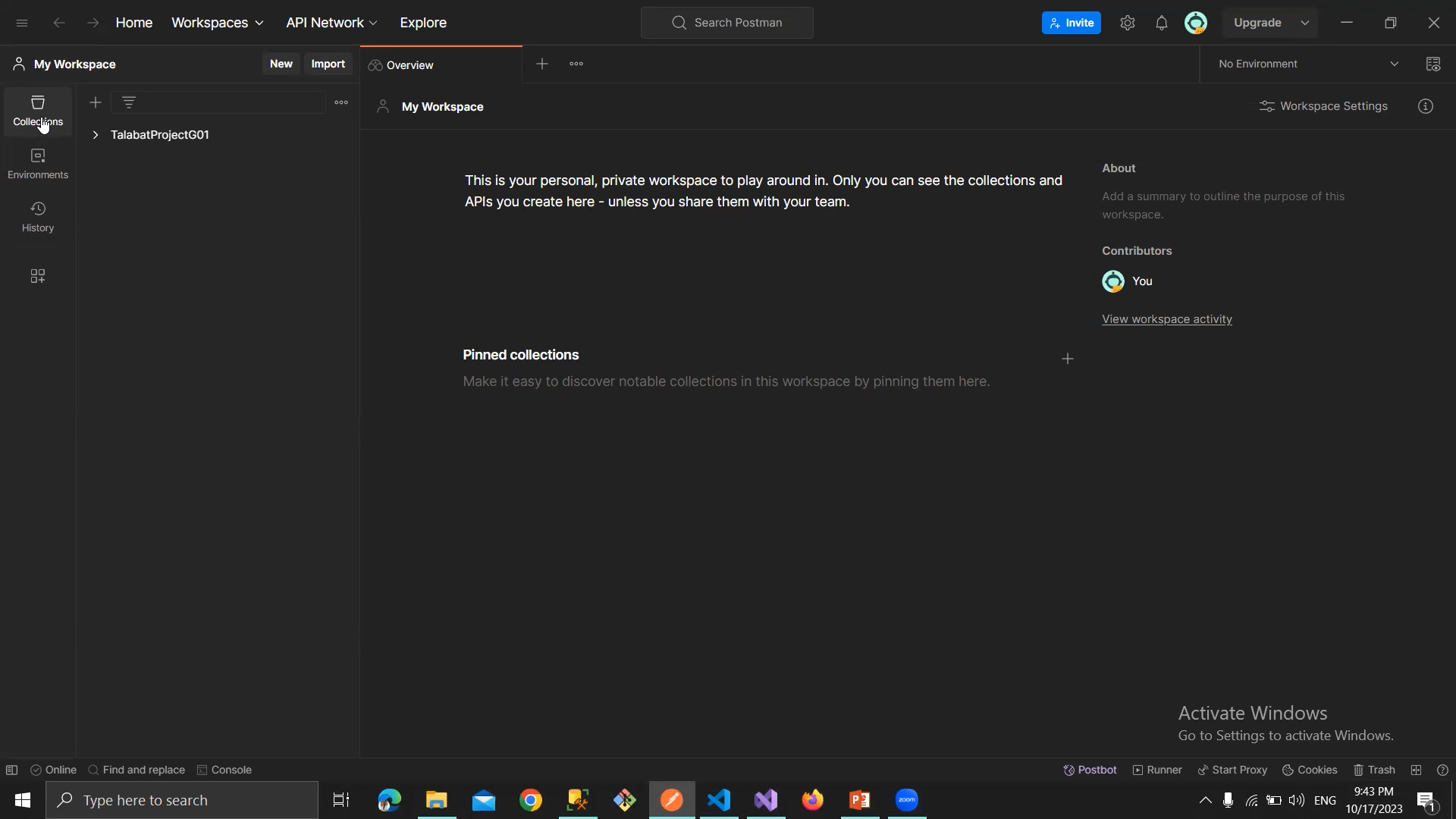
APIلما احتاج اضيف خارجي

على طول عشان انا مش محتاج الفيو اللي اللي موجوده في الكونترولرControllerBase >في الكونترولر بتاعي بورث من ال



|  |
| --- |
| **Web APIs بطبيعتها Stateless** |

|  |
| --- |
| كل Request مستقل تمامًا، ومفيش concept لـ "Redirect + TempData" |
| App.MapController>>هو هيفهم على طول انه هنشتغل بالكونترولرUseRouting,endPointفي 6 بدل اما اكتب |



Collection >>عباره عن البروجيكت بتاعي

)Default Getهقسم البروجيكت بتاعي لسيشنات واضيف الريكوست بتاعي (

Restful APIs:

[Route("api/[controller]")]

[ApiController]

**public class ProductsController : ControllerBase**

{

GetProducts() { } //get

GetProductsById(int id) { } //get(“id”)

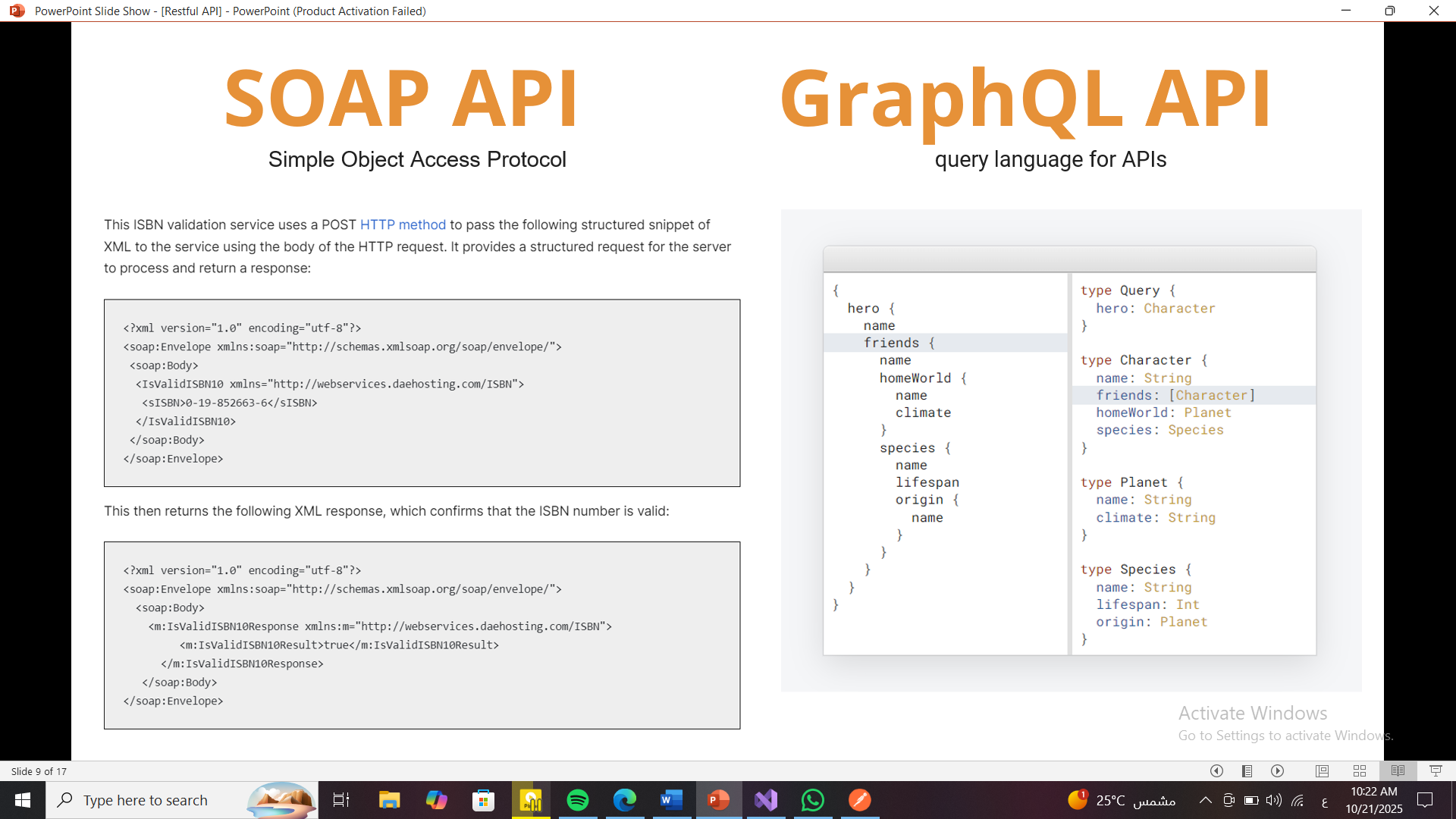
AddProduct(Product item) { }//post

UpdateProduct(Product item) { }//put

DeleteProduct(Product item) { }//delete

}

في نوعين غيره كنا بنستخدمه زمان:



soap>only xml

graphql>endpointعن طريق ريكويست واحد اقدر اكلم اكتر من

**Onion Layer Architecture:(pattern)**

**1-Domain Layer [Core Layer](center)**

**- Models [Class Represent Table In Database ]**

**- Core Of Project Not Implemented [Interfaces]**

**2-Repository Layer (M)**

* **(implementation)DbContext**
* **Repository [Part Of Business]>>احسن من التيرز بقى في الحته دي لان هنا اقددر اتعامل مع اكتر من كونتكس{هعمل لكل كونتكست ريبوزاتري لير ليها وانا هبقى حاطط الانترفيس بتاع الريبوزاتري في الدومين}**

**3-Services Layer**

* **Service [Rest Of Business]**

**Core Of Project Implemented [Implementation]**

**4-Presentation Layer**

**Core>baseEntity وهنعمل كلاس جوهاها فيها Entitiesهنبدا ونعمل فولدر لل**

\*\*MultipleActiveResultSets=true >>الكونيكشن بتاعي بيكلم كذا كويري في نفس الوقت

OnModelCreating: هيتحول لـ كابوس، كله متداخل وصعب تتتبعي فيه أي حاجة

protected override void OnModelCreating(ModelBuilder modelBuilder)

{

modelBuilder.Entity<Product>()

.Property(p => p.Name)

.IsRequired()

.HasMaxLength(100);

modelBuilder.Entity<Product>()

.Property(p => p.Price)

.HasColumnType("decimal(18,2)");

}

الطريقه الصح: <<Fluent API:

public class ProductConfiguration : IEntityTypeConfiguration<Product>

{

public void Configure(EntityTypeBuilder<Product> builder)

{

builder.Property(p => p.Name)

.IsRequired()

.HasMaxLength(100);

builder.Property(p => p.Price)

.HasColumnType("decimal(18,2)");

}

}

**تربطيه بـ DbContext:**

protected override void OnModelCreating(ModelBuilder modelBuilder)

{

modelBuilder.ApplyConfiguration(new ProductConfiguration());

}

أو بشكل أوتوماتيكي لكل الكيانات:

modelBuilder.ApplyConfigurationsFromAssembly(typeof(ApplicationDbContext).Assembly);

وبعدين هعمل الميجريشن هنزل التولز في اللير اللي هترن فيها tools(presentationLayer)

+startUpاتاكد انه

وبعدين هعمل الميجرين في الريبوزاتري

add-migration intialCreate -o Data/Migrations

>بحدد الاوت هيبقى فين

هعمل الابديت للداتا في البروجرام >>

var scope=app.Services.CreateScope();//Services Scoped

var services = scope.ServiceProvider;//DI

//loger Factory>>بتظهري الكسبشن

var LoggerFactory = services.GetRequiredService<ILoggerFactory>();

try

{

var DbContext = services.GetRequiredService<DbContext>();//ask clr to create obj

await DbContext.Database.MigrateAsync();//update DB

}

catch (Exception ex) {

var loggerFactory = LoggerFactory.CreateLogger<Program>();

}

>seeding>>هعمل داتا اجرب فيها