**یاسین شهریاری کوتک**

**پروژه پیاده سازی معماری تمیز با ویژوال استودیو**

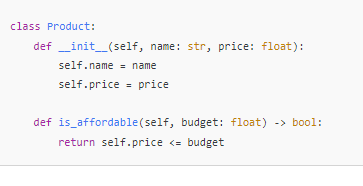
برای پیاده‌سازی کامل **معماری تمیز (Clean Architecture)** در یک پروژه نرم‌افزاری، مراحل زیر را دنبال کنید. این معماری به تفکیک مسئولیت‌ها در کد کمک می‌کند و وابستگی‌ها را به سمت جزئیات خارجی محدود می‌کند.

### ۱. لایه‌ها در معماری تمیز

معماری تمیز شامل چندین لایه است:

#### ۱.۱. لایه‌ی موجودیت‌ها (Entities)

این لایه شامل منطق اصلی کسب‌وکار و قوانین هسته‌ای است.  
مثال:



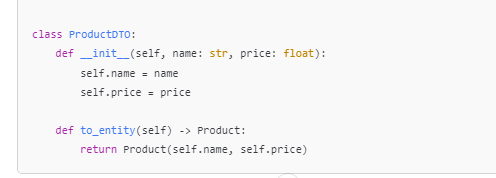
#### ۱.۲. لایه‌ی موارد استفاده (Use Cases)

این لایه وظایف کاربردی مشخص و وابسته به منطق کسب‌وکار را انجام می‌دهد. مثال:



#### ۱.۳. لایه‌ی رابط‌ها (Interface Adapters)

این لایه مسئول تطبیق مدل داده‌ها و تعاملات است. به‌عنوان مثال، مبدل‌هایی که داده‌ها را به قالب خاصی تبدیل می‌کنند. **مثال:**

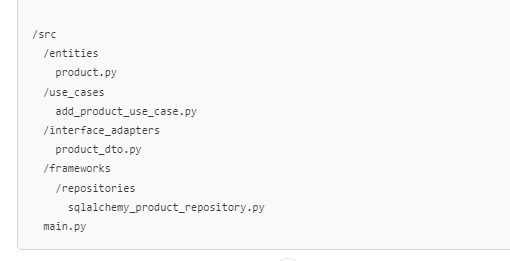


#### ۱.۴. لایه‌ی چارچوب‌ها و ابزارها (Frameworks & Drivers)

این لایه شامل پایگاه داده‌ها، رابط‌های کاربری، و سایر جزئیات زیرساختی است.  
مثال:



### 2. سازمان‌دهی پوشه‌ها

یک ساختار پیشنهادی برای پروژه: 

**۳. رعایت وابستگی‌ها**

وابستگی‌ها باید از بیرون به داخل باشد.

* **Use Cases** به **Entities** وابسته است، اما برعکس نیست.
* **Interface Adapters** می‌تواند به **Use Cases** وابسته باشد.
* **Frameworks** باید توسط **Dependency Injection** یا **Interfaces** قابل تغییر باشد.

**۴. تست‌ها**

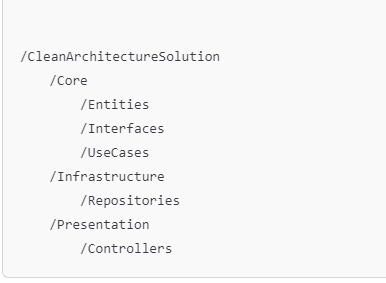
هر لایه را به طور مستقل تست کنید.

* **تست Use Cases** باید منطق تجاری را بدون توجه به دیتابیس آزمایش کند.
* از **Mocks** برای تست‌های سریع و مستقل استفاده کنید.

برای پیاده‌سازی **معماری تمیز (Clean Architecture)** در ویژوال استودیو (با استفاده از C#)، یک پروژه چندلایه را ایجاد کنید. این مراحل شما را به‌صورت عملی راهنمایی می‌کند.

**۱. ساختار پوشه‌ها و پروژه‌ها**

یک راهکار (Solution) ایجاد کنید و آن را به پروژه‌های مختلف تقسیم کنید:



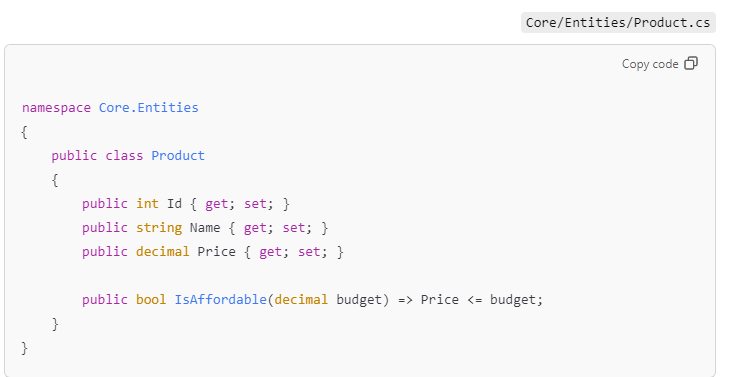
#### گام‌های ایجاد پروژه‌ها

1. **Core Project (Library)**  
   این پروژه منطق کسب‌وکار را نگهداری می‌کند.
2. **Infrastructure Project (Library)**  
   این پروژه به جزئیات زیرساختی مثل پایگاه داده می‌پردازد.
3. **Presentation Project (Web یا Console App)**  
   این پروژه به رابط کاربری می‌پردازد.

### ۲. پیاده‌سازی بخش‌ها

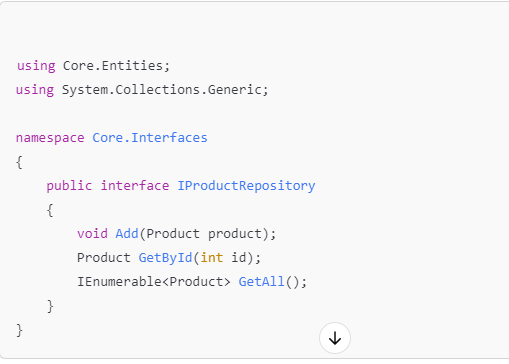
#### **Core Project**

##### ۲.۱. ایجاد یک Entity (کلاس محصول)



۲.۲. ایجاد یک Interface برای Repository

Core/Interfaces/IProductRepository.cs



۲.۳. ایجاد یک Use Case

Core/UseCases/AddProductUseCase.cs



Infrastructure Project

Infrastructure/Repositories/ProductRepository.cs

Presentation Project

Presentation/Program.cs



**۳. تنظیمات پروژه**

1. همه پروژه‌ها را در یک **Solution** قرار دهید.
2. **Core** و **Infrastructure** را به‌عنوان مرجع در **Presentation** اضافه کنید.

**۴. نکات مهم**

* **Dependency Injection** را برای مدیریت وابستگی‌ها به‌کار ببرید (در پروژه‌های واقعی).
* **Testing** را در هر لایه به‌طور مستقل اجرا کنید.