



SOLID-2

Home Work

.Net

۱. فایل تمرین را در پنل خود آپلود کنید.

۲. **title** فایل تمرین به صورت (نام تمرین+نام و نام خانوادگی) به انگلیسی باشد.

۳. در صورتی که سوال و یا ابهامی دارید در گروه چت تلگرامی بپرسید.

1. DIP – dependency inversion principle

مفاهیم زیر را تعریف کنید و ارتباط آن هارا باهم توضیح دهید :

- DIP
- IOC
- IOC Container
- Factory as IOC Container
- Dependency Injection
- Service LifeTimes in .Net

2. اصل DIP در طراحی ساده

یک برنامه بنویسید که اطلاعات محصولات را نمایش دهد. سه نوع منبع داده برای محصولات دارید:

۱. **Database:** دریافت اطلاعات از دیتابیس.

۲. **API:** دریافت اطلاعات از یک API.

۳. **File:** دریافت اطلاعات از یک فایل.

کار باید به گونه‌ای طراحی شود که:

- کلاس اصلی که اطلاعات محصولات را نمایش می‌دهد، وابسته به یک اینترفیس باشد، نه پیاده‌سازی خاص.
- منبع داده قابل تغییر باشد بدون اینکه نیازی به تغییر کد کلاس اصلی باشد.

3. استفاده از کلاس Factory

برنامه تمرین ۱ را گسترش دهید. این بار، از یک کلاس **Factory** استفاده کنید تا بر اساس یک تنظیمات یا ورودی مشخص، نوع منبع داده (Database, API, File) را انتخاب کند.

4. استفاده از IoC Container

برنامه تمرین ۲ را به گونه‌ای تغییر دهید که به جای ایجاد کلاس‌ها در کد، از یک IoC Container مثل **Microsoft.Extensions.DependencyInjection** برای مدیریت وابستگی‌ها استفاده کند.