

عنوان: استفاده از Delegates بجای اینترفیس‌ها در تزریق وابستگی‌ها

نویسنده: وحید نصیری

تاریخ: ۱۰:۴۳ ۱۳۹۲/۰۸/۲۸

آدرس: www.dotnettips.info

برچسب‌ها: Design patterns, Dependency Injection, IoC

عموما در معکوس سازی مسئولیت‌ها و واگذاری آن‌ها به لایه‌های دیگر، از اینترفیس‌ها استفاده می‌شود. روش دیگری را که در اینجا می‌توان بکار گرفته استفاده از Delegates است بجای اینترفیس‌ها. از این جهت که یک Delegate در عمل می‌تواند به صورت یک [Anonymous Interface](#) عمل کند. در بسیاری از مواردی که اینترفیس شما تنها از یک متد تشکیل می‌شود، می‌توان عملکرد آن‌را با یک Delegate جهت ساده سازی فرآیند تزریق وابستگی‌ها تعویض کرد.

یک مثال از تعویض اینترفیس‌های تک متدی با Delegates

```
public interface IAuthentication
{
    bool IsUserAuthenticated(string userName, string password);
}

public class AuthenticationService : IAuthentication
{
    public bool IsUserAuthenticated(string userName, string password)
    {
        return userName == "Vahid" && password == "123";
    }
}

public class LoginController
{
    private readonly IAuthentication _authentication;
    public LoginController(IAuthentication authentication)
    {
        _authentication = authentication;
    }

    // ...
}
```

مثال بسیار متداول فوق را در نظر بگیرید. در لایه سرویس برنامه، اینترفیس و کلاس پیاده سازی کننده منطق اعتبارسنجی کاربران را تدارک دیده‌ایم. نهایتاً جایی در سطحی بالاتر از این توانمندی قرار است استفاده شود. مثلاً در کلاس LoginController. نکته مهم اینترفیس IAuthentication، تک متدی بودن آن است. به همین جهت تعریف کلاس LoginController را به شکل زیر نیز می‌توان بازنویسی کرد:

```
public class LoginController
{
    private readonly Func<string, string, bool> _authenticationStrategy;
    public LoginController(Func<string, string, bool> authenticationStrategy)
    {
        _authenticationStrategy = authenticationStrategy;
    }

    // ...
}
```

در این حالت نیز کلاس LoginController استفاده کننده از Delegate تعریف شده، از نحوه و استراتژی اعتبارسنجی بی‌خبر است و پیاده سازی آن به کلاسی دیگر که قرار است از LoginController استفاده کند، واگذار می‌شود.