

عنوان: تزریق وابستگی‌ها و سناریوهای بسیار متعدد موجود

نویسنده: وحید نصیری

تاریخ: ۱۰:۷ ۱۳۹۲/۰۷/۲۲

آدرس: www.dotnettips.info

برچسب‌ها: Design patterns, Dependency Injection, IoC

تعدادی از پرکاربردترین حالت‌های تزریق وابستگی‌ها را در دوره جاری بررسی کردیم. برای مثال چگونه می‌توان تزریق وابستگی‌های یک کنترلر ASP.NET MVC را [خودکار کرد](#)، یا در [وب فرم‌ها](#) وضعیت چگونه است. اما در حین حل مسایلی از این دست، به سناریوهای بسیار متعددی برخورد خواهید خورد. برای مثال تزریق وابستگی‌های خودکار در یک کنترلر را آموختیم؛ در مورد فیلترها و Action Result‌های سفارشی چطور باید رفتار کرد؟ در WCF چطور؟ در هندلرهای وب فرم‌ها چطور؟ و بسیاری از حالات دیگر. البته تمام این موارد را توسط الگوی [Service locator](#) که شامل استفاده مستقیم از امکانات و هله سازی یک IoC Container، در کلاس مدنظر است، می‌توان حل کرد؛ اما باید تا حد امکان از این روش با توجه به اینکه خود IoC Container را تبدیل به یک وابستگی مدفون شده در کلاس‌های ما می‌کند، پرهیز نمود.

اگر به دنبال کتابخانه‌ای هستید که بسیاری از این سناریوها را پیاده سازی کرده است، کتابخانه Autofac پیشنهاد می‌شود. حتی اگر علاقمند به استفاده از آن نباشید، می‌توان از نحوه پیاده سازی‌های مختلف آن در مورد حالت‌های مختلف خودکار سازی تزریق وابستگی‌ها، ایده گرفت و سپس این کدها را با IoC Container مورد علاقه خود پیاده سازی کرد.

صفحه خانگی Autofac

<http://code.google.com/p/autofac>

<http://autofac.org>

بسته نیوگت

<http://www.nuget.org/packages/Autofac>

[محلی برای ایده گرفتن](#) مثلاً در مورد فیلترهای ASP.NET MVC

و در مورد نحوه استفاده از آن‌ها، نیاز است [آزمون‌های واحد این پروژه](#) را بررسی کنید و یا [مستندات پروژه](#) را مطالعه کنید. همچنین بررسی [لیست مستندات کلی آن](#) نیز بسیار مفید است

به صورت خلاصه، هرجایی در مورد تزریق وابستگی‌های خودکار جهت پرهیز از استفاده مستقیم از الگوی Service locator ایده‌ای نداشتید، سورس پروژه Autofac را بررسی کنید.

پ.ن.

[سایت bitbucket](#) امکان import کامل مخازن کد Google code را نیز دارد (در صورتیکه دسترسی شما به گوگل کد محدود است).