

همانطور که در مطلب " [NHibernate 3.0 و عدم وابستگی مستقیم به Log4Net](#) " عنوان شد، از اینترفیس جدید IInternalLogger آن می‌توان جهت ثبت وقایع داخلی NHibernate استفاده کرد. اگر در این بین صرفاً بخواهیم SQL های تولیدی را لاگ کنیم، خلاصه‌ی آن به صورت زیر خواهد بود:

```
public class LoggerFactory : ILoggerFactory
{
    public IInternalLogger LoggerFor(System.Type type)
    {
        if (type == typeof(NHibernate.Tool.hbm2ddl.SchemaExport))
            //log it
    }

    public IInternalLogger LoggerFor(string keyName)
    {
        if (keyName == "NHibernate.SQL")
            //log it
    }
}
```

یا کلید NHibernate.SQL باید پردازش شود (جهت ثبت SQL های کوئری‌ها) یا نوع NHibernate.Tool.hbm2ddl.SchemaExport جهت ثبت SQL ساخت ساختار جداول بانک اطلاعاتی باید بررسی گردد. [سورس](#) کامل این کتابخانه‌ی کوچک را [از اینجا](#) می‌توانید دریافت کنید. جهت استفاده از آن تنها کافی است چند سطر زیر به فایل app.config یا web.config برنامه‌ی شما اضافه شوند:

```
<appSettings>
    <add key="nhibernate-logger" value="NH3SQLLogger.LoggerFactory, NH3SQLLogger" />
</appSettings>
```

کلید nhibernate-logger ، به صورت مستقیم توسط NHibernate بررسی می‌شود و صرف نظر از اینکه از کدامیک از مشتقات NHibernate استفاده می‌کنید، با تمام آن‌ها کار خواهد کرد. لازم به ذکر است که اگر برنامه‌ی شما از نوع ASP.NET است، این کتابخانه اطلاعات را در پوشه‌ی استاندارد App\_Data ثبت خواهد کرد؛ در غیراینصورت فایل‌ها در کنار فایل اجرایی برنامه تشکیل خواهند شد.

## نظرات خوانندگان

نویسنده: A

تاریخ: ۲۳:۳۳:۴۵ ۱۳۸۹/۱۰/۰۸

شاید این موضوع بی ربط به پست جاری باشه. از این بابت عذر خواهی می‌کنم.

اینطور که در کدهای شما دیدم شما اغلب از `CodeStyle` یی شبیه `Developer` های سیستم‌های میکروسافت استفاده میکنید. که بارزترین تفاوت آن استفاده از `Underline` در ابتدای فیلدهای تعریف شده در کلاس است. حتماً می‌دانید که `StyleCop` نیز که توسط خود میکروسافت ارائه شده، این روش را رد می‌کند و می‌گوید `Underline` زیبایی را کاهش می‌دهد و به جای آن از `this`. استفاده شود که محاسن دیگری نیز دارد.

البته به قول نویسنده `StyleCop` اینکه بچه‌های میکروسافت اینطور کد می‌نویسند یک دلیل منطقی دارد؛ و آن این است که آنها بیشتر برنامه‌نویسان `C++` بوده‌اند و در `C++` شبیه این سبک بیشتر رایج است (درست است که این روش، روش مجارستانی نیست ولی حداقل استفاده از `this` به هیچ عنوان در `C++` رایج نبود).

خوب، سوال اینجاست. نظر شما چیست؟

نویسنده: وحید نصیری

تاریخ: ۲۳:۵۰:۴۸ ۱۳۸۹/۱۰/۰۸

خوانایی استفاده از `Underline` برای معرفی `private field names` از حالت استفاده از `this` بیشتر است (فقط با یک نگاه مشخص می‌شود). همچنین امکان فراموش شدن استفاده از `this` هم ممکن است باشد که این مورد سبب بروز احتمال تداخل متغیرهای یک متد با نمونه‌ای که به صورت `private field` تعریف شده می‌گردد. فقط موردی را که باید در نظر داشت، یکپارچگی است. یعنی در کل کدهای شما یک روش باید وجود داشته باشد. همچنین چون من برای بازبینی کدها یکبار از `resharper` هم استفاده می‌کنم، توصیه `resharper` استفاده از `Underline` هست و `this` را به صورت زاید (`redundant`) معرفی می‌کند.

نویسنده: A

تاریخ: ۰۲:۳۴:۳۷ ۱۳۸۹/۱۰/۰۹

البته یک چیزی خیلی برایم جالب است که ایشان به `Underscore` معتقد است:  
<http://10rem.net/articles/net-naming-conventions-and-programming-standards---best-practices>

ولی در MSDN ذکر شده که

"Do not apply a prefix to field names or static field names"

منبع: پاراگراف آخر [http://msdn.microsoft.com/en-us/library/ta31s3bc\(v=vs.71\).aspx](http://msdn.microsoft.com/en-us/library/ta31s3bc(v=vs.71).aspx)