

عنوان: روش نامگذاری Smurf ای!

نویسنده: وحید نصیری

تاریخ: ۱۱/۰۶/۱۳۹۱

آدرس: www.dotnettips.info

برچسب‌ها: C#, Design patterns, Naming

اگر به یک سری از کتابخانه‌ها دقت کنید، تمام کلاس‌های آن‌ها دارای یک پیشوند تکراری هستند؛ مثلاً **Smurf** **XMLDataRow**، **Smurf** **XmlElement** و الی آخر در مورد تمام کلاس‌های موجود در پروژه. به این رویه «[Smurf Naming Convention](#)» گفته می‌شود! در این نوع کتابخانه‌ها زمانیکه کاربری بر روی دکمه‌ای کلیک می‌کند، **Smurf** **AccountView** اطلاعات **Smurf** **AccountDTO** را به **Smurf** **AccountController** منتقل می‌کند. در ادامه از خاصیت **Smurf** **ID** دریافتی، مقدار **Smurf** **OrderHistory** دریافت شده و به **Smurf** **HistoryReportingView** جهت نمایش ارسال خواهد شد. اگر استثنای **Smurf** **ErrorEvent** رخ دهد، توسط **Smurf** **ErrorLogger** در فایلی به نام **log/ smurf / smurf log.log** ثبت خواهد شد.

کلمه **Smurf** هم از شخصیتی کارتونی به همین نام اخذ شده است که در زبان مخصوص آن‌ها اکثر افعال و نام‌ها از کلمه **Smurf** مشتق می‌شود! برای مثال در مورد ماهیگیری کردن در یک رودخانه عنوان می‌کنند «**We're going smurfing on the River Smurf**» **today**.



خوب، چکار باید کرد؟ روش صحیح معرفی نام یک شرکت در حین طراحی و نامگذاری کلاس‌های یک کتابخانه چیست؟ در مطلب بسیار جامع و عالی «[اصول و قراردادهای نام‌گذاری در دات‌نت](#)» عنوان شده است که اساس نام‌گذاری فضاهای نام باید از قاعده زیر پیروی کند:

```
<Company>.<Technology|Product|Project>[.<Feature>][.<SubNamespace>]
```

مثلاً میکروسافت یکبار فضای نام **Microsoft.Reporting.WebForms** را تعریف کرده است و ... همین! دیگر به ابتدای هر کلاسی در این کتابخانه، پیشوند **Microsoft** یا **MS** و امثال آن اضافه نشده است تا بر روی اعصاب و روان استفاده کننده تاثیر منفی داشته باشد.

نظرات خوانندگان

نویسنده: رحمت اله رضایی
تاریخ: ۱۸:۳۸ ۱۳۹۱/۰۶/۱۱

مشکل اینجاست که در پروژه ای، کلاسهایی هم اسم کلاسهای دات نت داشته باشیم. همیشه باید فضای نام را در ابتدای کلاسها نوشت.
مثلا فرض کنید کنترلرهای TextBox و Button و ... را در یک پروژه وب فرم یا ویندوز فرم سفارشی کرده باشیم. در این حالت چکار باید بکنیم؟

نویسنده: وحید نصیری
تاریخ: ۱۹:۲۰ ۱۳۹۱/۰۶/۱۱

- برنامه [FxCop](#) می‌تونه اسمبلی‌های شما رو آنالیز کنه و دقیقا گزارش بده که چه مواردی هم نام کلاس‌های پایه دات نت هستند و بهتر است تغییر نام پیدا کنند. بنابراین به این صورت می‌تونید خیلی سریع حجم بالایی از کدها رو بررسی و رفع اشکال کنید.
- به علاوه زمانیکه طراح شما هستید، محدودیتی در نامگذاری نهایی وجود ندارد. مثلا نام کلاس مشتق شده را NumericTextBox قرار دهید و مواردی مانند این که بیانگر عملکرد سفارشی و ویژه کلاس مشتق شده جدید هستند:

```
public class RequiredTextBox : TextBox
```