عنوان: NHibernate 3.0 و ارائه‌ی جایگزینی جهت ICriteria API
نویسنده: وحید نصیری
تاریخ: ۱۳۸۹/۰۹/۳۰ ۱۱:۱۵:۰۰
آدرس: www.dotnettips.info
برچسب‌ها: NHibernate

ICriteria API در NHibernate پیاده سازی الگوی Query Object است. مشکلی هم که این روش دارد استفاده از رشته‌ها جهت ایجاد کوئری‌های متفاوت است؛ به عبارتی Type safe نیست. ایرادی هم به آن وارد نیست چون پیاده سازی اولیه آن از جاوا صورت گرفته و مباحث Lambda Expressions و Extension Methods هنوز در آن زبان به صورت رسمی ارائه نشده است (در JDK 7 تحت عنوان Closures قرار است اضافه شود). NHibernate 3.0 از ویژگی‌های جدید زبان‌های دات نتی جهت ارائه‌ی محصور کننده‌ای Type safe حول ICriteria API استاندارد به نام QueryOver API سود جسته است. این پیاده سازی بسیار شبیه به عبارات LINQ است اما نباید با آن اشتباه گرفته شود زیرا LINQ to NHibernate یک ویژگی دیگر جدید، یکپارچه و استاندارد NHibernate 3.0 به شمار می‌رود.
برای نمونه در یک ICriteria query متداول، فراخوانی‌های ذیل متداول است:

```
.Add(Expression.Eq("Name", "Smith"))
```

اکنون شما در NHibernate 3.0 می‌توانید دستورات فوق را به صورت ذیل وارد نمائید:

```
.Where<Person>(p => p.Name == "Smith")
```

مزیت‌های این روش (strongly-typed fluent API) به شرح زیر است:
- خبری از رشته‌ها جهت استفاده از یک خاصیت وجود ندارد. برای مثال در اینجا خاصیت Name کلاس Person تحت کنترل کامپایلر قرار می‌گیرد و اگر در کلاس Person تغییراتی حاصل شود، برای مثال Name به LName تغییر کند، برنامه دیگر کامپایل نخواهد شد. اما در حالت ICriteria API یا باید به نتایج حاصل از Unit testing مراجعه کرد یا باید به نتایج بازخورد کاربران برنامه مانند: "باز برنامه رو تغییر دادی، یکجای دیگر از کار افتاد!" دقت نمود!
- اگر در حین ویرایش کلاس Person از ابزارهای Refactoring استفاده شود، تغییرات حاصل به صورت خودکار به تمام برنامه نیز اعمال خواهد شد. بدیهی است این اعمال تغییرات تنها در صورتی میسر است که خاصیت مورد نظر به صورت رشته معرفی نگردیده و ارجاعات به اشیاء تعریف شده به سادگی قابل parse باشند.
- در این حالت امکان بررسی نوع خواص تغییر کرده نیز توسط کامپایلر به سادگی میسر است و اگر ارجاعات تعریف شده به نحو صحیحی از این نوع جدید استفاده نکنند باز هم برنامه تا رفع این مشکلات کامپایل نخواهد شد که این هم مزیت مهمی در نگهداری ساده‌تر یک برنامه است.
- با بکارگیری Extension methods و پیاده سازی Fluent API جدید، مدت زمان یادگیری این روش نیز به شدت کاهش یافته، زیرا Intellisense موجود در VS.NET بهترین راهنمای استفاده از امکانات فراهم شده است. برای مثال جهت استفاده از ویژگی جدید QueryOver به سادگی می‌توان پس از ساختن یک session جدید به صورت زیر عمل نمود:

```
IList<Cat> cats = session.QueryOver<Cat>().Where(c => c.Name == "Max").List();
```

در اینجا اگر متدهای نمایش داده شده توسط Intellisense را دنبال کنیم دیگر حتی نیازی به مراجعه به مستندات QueryOver در مورد اینکه چه متدها و امکاناتی را فراهم کرده است نیز نخواهد بود.

جهت مشاهده‌ی معرفی کامل آن می‌توان به مستندات NHibernate 3.0 مراجعه کرد.

نظرات خوانندگان

نویسنده: Meysam
تاریخ: ۱۳۸۹/۰۹/۳۰ ۱۶:۰۳:۵۲

Oren Eini یک روش شبیه به این برای INPC هم داده که جالبه

نویسنده: A
تاریخ: ۱۳۸۹/۰۹/۳۰ ۱۶:۱۰:۰۹

گذشته از بحث NHibernate، اینطوری که پیداست واقعاً جاوا Follower سی شارپ شده!!!

نویسنده: ghafoori
تاریخ: ۱۳۸۹/۱۰/۰۱ ۱۸:۵۶:۳۴

سلام آقای نصیری
من از کد زیر برای جستجو در بانکم بوسیله nhibernate ویرایش 3 استفاده می کنم اما خطا میدهد

```
(Dim Query = ServerRepo.Find(Function(x) x.Country Is Country And x.ServerName Is ServerName
```

خطای زیر

```
Unable to cast object of type 'NHibernate.Hql.Ast.HqlBitwiseAnd' to type
.''NHibernate.Hql.Ast.HqlBooleanExpression
```

وقتی بجای Is از = استفاده می کنم خطای زیر را می دهد

```
(Int32 CompareString(System.String, System.String, Boolean
```

فکر می کنید مشکل از کجاست با تشکر

نویسنده: وحید نصیری
تاریخ: ۱۳۸۹/۱۰/۰۱ ۱۹:۵۸:۳۴

مشکل مرتبط است با زبان VB.NET ، جهت توضیحات بیشتر و ارائه راه حل (که باید کمی کدهای اصلی NHibernate را ویرایش (جایگزینی VBStringComparisonExpression با BinaryExpression) و سپس کامپایل کنید) این دو مقاله را مطالعه کنید:
(+) و (+)

نویسنده: ghafoori
تاریخ: ۱۳۸۹/۱۰/۰۲ ۱۲:۴۲:۳۱

اقای نصیری بابت راهنمایی خیلی متشکر
من ترسیدم برم سراغ کدهای nhibernate و کلا پروژه ام را از وی بی به سی شارپ تغییر دادم مشکل هم حل شد
مخصوصا با این تبدیل کننده شرکت تلریک
http://converter.telerik.com/batch.aspx
سریع تونستم هسته پروژه که با nhibernate سرو کار داره را از وی بی به سی شارپ انتقال بدم
باز هم بابت راهنمایی ممنونم

نویسنده: وحید نصیری
تاریخ: ۱۳۸۹/۱۱/۰۸ ۱۵:۰۸:۱۳

یک مورد شبیه به این QueryOver را هم اینجا می‌توانید پیدا کنید:

nhflowquery blog , nhflowquery