

**Stopařův průvodce po ADO.NET a LINQ
aneb
Foxyřův průvodce po práci s daty v .NET**

Milan Kosina

Obsah

Stopařův průvodce po ADO.NET a LINQ aneb Fogařův průvodce po práci s daty v .NET	1
Milan Kosina	1
Úvod	4
Co je nutné nainstalovat	4
O čem se budeme bavit.....	4
LINQPad.....	5
ADO.NET	7
ADO.NET – Připojení.....	7
Připojení k SQL Serveru	7
ADO.NET – základní stavební prvky.....	8
ADO.NET – SqlDataReader	9
Dotaz, který vrací 1 hodnotu	9
Dotaz, který vrací 1 tabulku (ale ne úplně)	9
Dotaz s parametry	10
Vlastní zpracování dotazu	11
Hodnoty NULL	13
Dotaz vracející více výsledků	14
Struktura vrácené tabulky	15
Dotaz, který nevrací tabulku	16
ADO.NET – DataSet, DataTable a další.....	16
Vytvoření DataSetu a DataTable	17
Vkládání dat do DataTable	17
Změna dat v DataTable.....	19
Rušení dat v DataTable.....	21
Naplnění DataTable nebo DataSetu daty ze SQL Serveru pomocí SqlDataReaderu	22
Naplnění DataTable nebo DataSetu daty ze SQL Serveru pomocí SqlDataAdapteru.....	23
Aktualizace dat na SQL Serveru pomocí SqlDataAdapteru	25
DataTable – práce s řádky a sloupci	30

DataTable – třídění, hledání a filtrování.....	31
DataTable a DataView – aneb to nejdůležitější!	33
ADO.NET – transakce	36
ADO.NET – „Typed DataSet“	37
LINQ.....	38
Přehled výhod LINQu.....	38
Generování zdrojového programu pro LINQ to SQL a SQLMetal.....	39
DataContext.....	40
První příklad na LINQ to SQL	41
Přehled lambda operátorů	43
Filtrování dat	43
Projekce	44
Spojování více tabulek.....	49
Třídění	51
Agregace (seskupování).....	51
Další příklad	52
Aktualizace dat	53
Transakce.....	55
Ošetření chyb při aktualizaci	56
Uložené procedury	57
Vytvoření malé aplikace	59
Vytvoření stejné aplikace ve Visual FoxPro a v C# s využitím LINQ.....	59
Stejná aplikace v C# s využitím ADO.NET	66
Vytvoření webové aplikace (Web Forms) v C# s využitím LINQ.....	68
Závěr	74
Použitá literatura.....	74

Úvod

Cílem této brožurky je poskytnout pomoc programátorům ve Visual FoxPro, kteří hledí s nedůvěrou na ostatní programovací nástroje a jsou přesvědčeni, že žádný jiný nástroj jim nemůže poskytnout takový komfort při práci s daty jako „stará dobrá Foxka“ ☺. Budu se snažit Vám (programátorům ve VFP) pomoci tím, že budu porovnávat způsob práce ve Visual FoxPro a v .NET. Budu používat jazyk C#, ale pokud byste chtěli používat VB.NET, nezoufejte – práce ve VB.NET je stejná jako v C#, jen syntaxe se trochu liší.

Z výše uvedeného je zřejmé, že pro čtení dalších kapitol je nezbytná znalost práce s Visual FoxPro. **Není** ale potřeba znát C#, neboť příklady budou psány tak, aby jim bylo možné rozumět i bez znalosti tohoto jazyka.

Co je nutné nainstalovat

Mým cílem bylo, abyste si mohli všechny příklady sami vyzkoušet – a to s minimem vynaložené námahy. Pro naše povídání proto bude třeba nainstalovat pouze následující programy (vše se dá pořídit zdarma):

- Data budou uložena na SQL Serveru, takže je nezbytné mít libovolnou verzi SQL Serveru - buď Express verzi (která je zdarma) nebo plnou verzi. Samozřejmě je možné použít i MSDE. Vzhledem k tomu, že většina z nás už nějakou (třeba Express) verzi na nějakém CD určitě má, URL neuvádím.
- Potřebujeme mít stejnou databázi pro naše příklady. Zkontrolujte proto, zda vaše instalace SQL Serveru obsahuje databázi NorthWind. Pokud ji neobsahuje, stáhněte si ji z <http://www.microsoft.com/downloads/details.aspx?FamilyID=06616212-0356-46A0-8DA2-EEBC53A68034&displaylang=en> (1,5 MB) a nainstalujte ji podle pokynů v dokumentaci.
- Pro poslední kapitolku („Vytvoření malé aplikace“) budeme potřebovat Visual Studio 2008. Pro Vaše první pokusy bude určitě stačit Express verze, která je zdarma (<http://www.microsoft.com/express/vcsharp/>). Pro první dvě kapitoly („ADO.NET“ a „LINQ“) je třeba mít nainstalován minimálně .NET Framework 3.5 SP1, který lze stáhnout z <http://download.microsoft.com/download/2/0/e/20e90413-712f-438c-988e-fdaa79a8ac3d/dotnetfx35.exe>. (**Pozor!** Je to 230 MB.) Musí se jednat o verzi 3.5 - bude nás totiž zajímat LINQ.
- Vzhledem k tomu, že mým cílem je usnadnit vám pokusy s .NET, budeme v prvních dvou kapitolách pracovat v programu LINQPad, který je zdarma a můžete si ho stáhnout z <http://www.linqpad.net/>. Více si o tomto skvělém programu povíme v samostatné kapitole.

O čem se budeme bavit

Společně se podíváme na dva způsoby práce s daty: ADO.NET a LINQ.

ADO.NET (Active Data Objects for .NET): Umožňuje objektovou práci s relačními daty na nejnižší úrovni. *Příklad pro foxaře:* Odpovídá zhruba SQL-Pass-Through a vzdáleným pohledům. Rozdíl je v tom, že výsledkem není cursor, ale objekt vytvořený z nějaké třídy (podstatně bohatší než tabulka).

LINQ (Language INtegrated Query): Jazyk, který lze v .NET použít pro dotazy a aktualizace dat bez ohledu na to, odkud data pocházejí. LINQ To SQL interně používá ADO.NET. *Překlad pro foxaře:* Umožňuje klást dotazy psané stejným jazykem jak na data na SQL Serveru, tak na data v XML souboru, tak i na procesy běžící ve Windows apod.

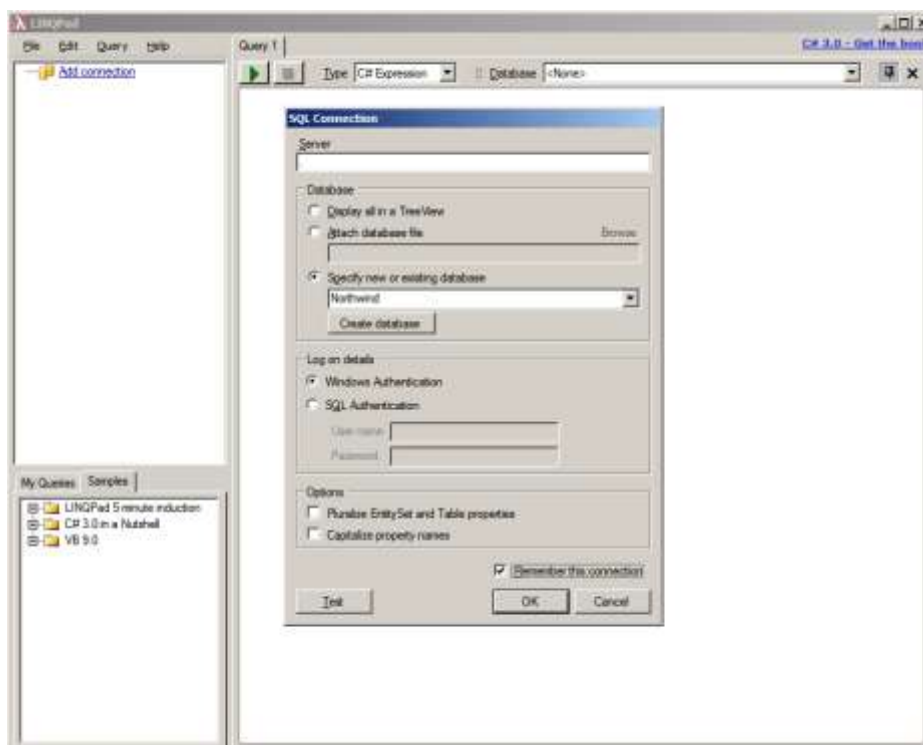
Microsoft v současnosti více prosazuje pro práci se SQL Serverem a jinými databázemi svůj Entity Framework, který je složitější než LINQ, a slibuje, že ve verzi 2.0, na které pracuje, bude možno pracovat s databázemi stejně snadno jako v LINQ To SQL. Uvidíme, jak to dopadne, nicméně i v Entity Frameworku můžete používat LINQ To Entities... Navíc existují i konkurenční projekty jako nHibernate (s LINQ to nHibernate), LLBLGen Pro a další ORM (Object/Relation mapper).

LINQPad

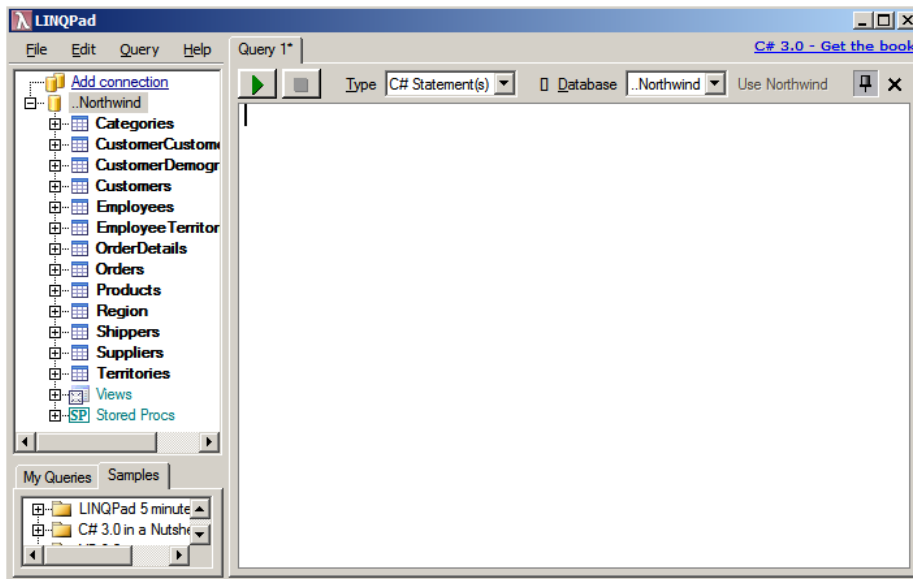
LINQPad je skvělý malý (2 MB) program, který je zcela zdarma. Dají se v něm snadno zkoušet kousky programů, aniž bychom museli mít nainstalované Visual Studio. To bude pro nás výhodné v prvních kapitolách, protože nebudeme muset psát „omáčku“ kolem testovaného kousku programu.

V závěrečné kapitole si ukážeme výhody Visual Studia na skutečných malých prográmcích.

Nejdůležitější je zadat správně připojení k SQL Serveru. Později už si bude LINQPad toto připojení pamatovat. Spusťte tedy LINQPad, klikněte na „Add connection“ a vyplňte dialogové okno tak, jak je uvedeno (tečka v Server znamená, že se jedná o server na tomto PC - je to stejné jako „(local)“):

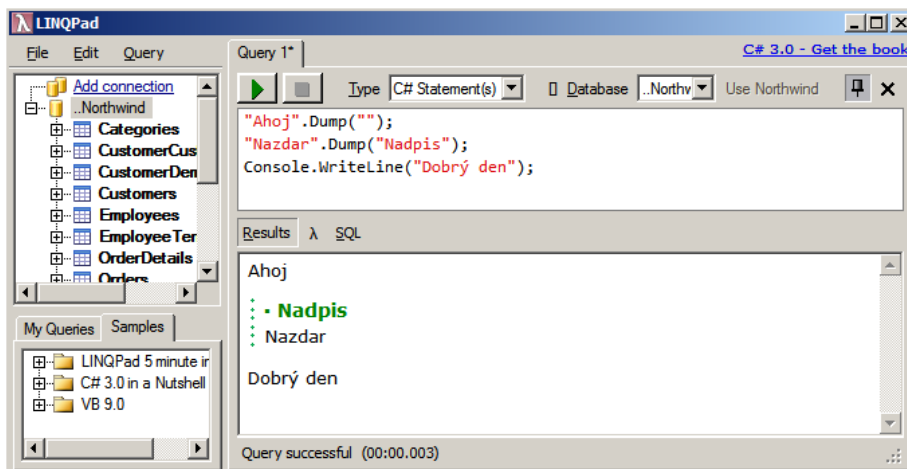


Změňte potom ještě Type na „C# Statement(s)“ a Database na „..Northwind“ a jste připraveni programovat.



Příkazy končí v C# středníkem. Pro jejich spuštění klikneme na zelenou šipku nebo stiskneme F5. Pro pomocné výpisy budeme používat dva způsoby:

- Prakticky za cokoli lze (pozor - pouze v LINQPadu!) připsat `.Dump()` nebo `.Dump("Nadpis")`. LINQPad pak zobrazí vše, co o daném objektu zjistí – případně včetně nadpisu (všimněte si svislé čáry, která spojuje nadpis a výpis).
- `Console.WriteLine()` nám umožňuje něco podobného jako „?“ ve Foxce.



Od této chvíle už budu ve screenshotech ukazovat pouze pravou stranu LINQPadu.

Mimochodem, LINQPad napsali autoři knihy „C# 3.0 in a nutshell“, kterou vám velmi doporučuji (viz „Použitá literatura“).

Malá poznámka: V současnosti je na webu novější verze LINQPadu než ta, se kterou jsem vytvářel tyto screenshoty.

ADO.NET

ADO.NET zhruba odpovídá tomu, jak je se SQL Serverem schopna pracovat Visual FoxPro. Nebudeme tedy dlouho teoretizovat a postupně si ukážeme, co je a co není v ADO.NET možné.

ADO.NET – Připojení

Připojení k SQL Serveru

Ve Visual FoxPro bychom k připojení použili

```
connString = ;  
"DRIVER=SQL Server;SERVER=(local);Trusted_Connection=Yes;Database= Northwind;"  
nHandle = SQLSTRINGCONNECT(connString)
```

a k odpojení

```
SQLDISCONNECT(nHandle)
```

Vzhledem k tomu, že proces otevírání trvá dlouho, patrně otevřete spojení na začátku programu a zavřete na konci.

V .NET se otevírání a zavírání zapíše podobně:

```
string connString =  
"Data Source=(local);Initial Catalog=Northwind;Integrated Security=True;";  
  
using (SqlConnection conn = new SqlConnection(connString))  
{  
    conn.Open();    // Otevřeme spojení  
    conn.Dump();    // Pracujeme. Zde si místo toho vypíšeme vlastnosti  
    conn.Close();   // A zavřeme spojení!
```