



Střední škola pedagogická, hotelnictví a služeb, Litoměřice, příspěvková organizace

Předmět:	Vývoj aplikací
Téma:	Visual Studio
Vyučující:	Ing. Milan Káža
Třída:	EK4
Hodina:	4-6
Číslo:	V/5

Programování v jazyce C a C#

Visual Studio

12. Databáze a práce s ní

12.1. Pojem databáze

Databáze je uspořádaný systém dat uložený na jednom médiu, odkud se informace načítají upravují a znovu ukládají. Databázi je možné přirovnat ke kartotéce, kde jsou uloženy informace např. o pacientech.

V C# se pro spolupráci s databázemi používá Server Microsoft SQL a pro přístup k databázi se používá **Data.SqlClient**. Pro přístup k datbázím se dále používá **System.Data.OleDb**, to jsou třídy používané poskytovatelem **OleDbProvider**. Ještě existují další poskytovatelé jako je ODBC, FireBird nebo MySQL.

12.1.1. Připojení k databázi

V rámci Visual Studia je možné vytvářet databáze a připojení k datbázi, pro které se použije třída **Connection** a jako argument konstrukturu se použije tzv. „připojovací řetězec“, který definuje cestu k databázi a její připojení.

Vytvoření připojovacího řetězce

Připojení k SQL serveru se zprostředkovává třídou **System.Data.SqlClient.SqlConnection** a je nutné znát některé údaje pro připojení. Připojovací řetězec se připojuje konstruktorem **ConnectionString**. Připojovací řetězec může obsahovat řadu nastavení pro připojení jako je název serveru, heslo atd.

Pozor: *Nezaměňujte MS SQL z My SQL. MS SQL je databáze Microsoftu v rámci Visual Studia, ale My SQL je databáze pro server APACHE. Příkazy pro databáze jsou totožné, ale připojení je různé.*

Connection - přístup k databázi

Aby bylo možné s databází pracovat, je třeba nejprve vytvořit připojení. V závislosti na používaném serveru se vybere typ připojení. Standartně se používá **Microsoft SQL Server**. Proto se pro volání použije typ **SqlConnection** z jmeného prostoru **System.Data.SqlClient**. Dále jen nutné **ConnectionString**, řetězec používaný pro připojení k databázovému serveru, ve kterém se uvádí název databáze, login, heslo a další parametry. Ve Visual Studiu není nutné toto znát. Stačí nastavit cestu: při zobrazení v designer (**View -> Designer**) z **Toolboxu (View -> Toolbox - Data)** a vybrat prostředek s názvem **SqlConnection** a vložit do formuláře.

Tím se vytvoří nový objekt **SqlConnection**. Zdrojový kód každého vkládaného prvku je možné zobrazit poklepnutím pravým tlačítkem myši na příslušný prvek a vybráním možnosti **View Code**. Po vytvoření objektu se vytvoří část kódu

Zde je ukázka kódu

```
this.sqlConnection = new System.Data.SqlClient.SqlConnection();
```

kde sqlConnection je jméno instance.

Celý kód pro připojení k jednomu ze serverů v tomto případě MS SQL je a k databázi Northwind(tj. vzorová databáze) se vytvoří tímto způsobem:

```
string popis= "server=localhost;database=Northwind;uid=sa;pwd=sa";
SqlConnection spojeni=null;
try{
    spojeni= new SqlConnection(popis);
    spojeni.Open();
}
finally
{
    if (spojeni !=null) spojeni.Close();
}
```

Připojovací řetězec obsahuje URL server (**localhost**), tento server je možné mít umístěny na místním počítači, není tedy nutné mít připojení k internetu a neustále navazovat se serverem spojení přes internet. Dále obsahuje připojovací řetězec uživatelské jméno (**uid**), které dává přístup k databázi a uživatelské heslo (**pwd**), které provádí ověření zda uživatel oprávnění do databáze vstupovat.

Po navázání spojení přes metodu **Open()** se databáze také ukončuje metodou **Close()**.

12.2. Systém ADO.NET

ADO.NET (Microsoft ActiveX Data Objects .NET) představuje množinu tříd nabízejících služby pro přístup k datům a tvorbu databázových aplikací. Data jsou informace uložené v databázích. Ať již se jedná o data v databázích například na Microsoft SQL Serveru či data zpřístupněná přes OLE DB nebo XML. Mezi jeho přednosti patří především jednoduchý způsob použití, rychlost při zpracování a další. Stačí vytvořit spojení se serverem, s kterého se budou data získávat.

ADO.NET ale nemusí pracovat pouze s databázemi na nějakém serveru. Bylo navrhováno současně s XML třídami v prostředí .NET Framework. Také díky tomu je možno data načítat i ve formátu XML nebo data zapisovat jako XML soubory spolu s definičním souborem XSD definujícím schéma dané databáze.

Nástroje ADO.NET byly navrženy tak, aby se oddělil způsob přístupu k datům od manipulace s daty. K první skupině patří .NET Framework data provider obsahující množinu komponent zahrnujících podmnožiny Connection (připojení), Command (množinu příkazů pro vybrání dat), DataReader (načítání dat) a DataAdapter (adaptér pro připojení k databázi). K druhé skupině se řadí mimo jiné objekt DataSet (skládající se z objektů DataTable, DataRow...). Jedná se o objekty uchovávající data načtená z databázi. Tyto objekty mohou s daty pracovat stejně jako s daty v databázi.

Kontrolní otázky

1. Vysvětlete pojem databáze a jak se připojuje k serveru.
2. Objasněte ADO.NET

Použitá literatura:

Miroslav Virius: od C++ k C# ,1.vydání, KOOP České Budějovice, 2002

Eric Gunnerson: Začínáme programovat v C# , 1. vydání, Computer Press, 2001

John Sharp, Jon Jagger: Microsoft Visual C# .NET krok za krokem, 1. vydání, Knihy.iDnes

Miroslav Virius: C# Hotová řešení, 1. vydání, Computer Press, 2006

Amadeo Mareš: 1001 tipů a triků pro C#