

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

## 《现代软件开发——使用.NET与C#》系列第七讲

### ➤ **ADO.NET: 数据库编程**

俞晖  
MSDN讲师  
dolphin@vip.163.com

# 《现代软件开发——使用.NET与C#》 系列

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

- 回顾上节课的内容  
(*WinForms*)
- 本节课讲述在  
**C#**中的数据库  
编程方面的问题
- **LEVEL 200**
- 基础:
  - 初级编程
  - **DB&SQL**

Date	Topic
2005年8月1日	<i>WinForms</i>
2005年8月16日	<i>ADO.NET</i>
2005年9月	Multi-tier Application Design
.	.

# 关于数据库编程的观点

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

- 幼稚的开发

- 代码完全放置在 **GUI** 中— 差，太差了
- 代码完全使用明确的**SQL**（不用存储过程）— 慢，太慢了
- 我们将在未来的**Webcast**中，应用多层结构重新设计今天讲的代码，

- 手写所有的代码

- 很多工具都可以帮助你直接生成代码，如 **VS2003, VS2005** 中的数据窗体，但今天希望大家了解这些代码。

# 今天的例子

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

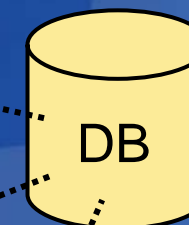
- 简单的关系数据库：

	CID	FirstName	LastName	CreditLimit	AcctBalance	DateOfEntry	Notes
	1	Jim	Bag	\$1,000.00	\$0.00	01-Jan-98	works at the gym
	3	Kathie	O'Dahl	\$9,999.99	\$0.00	02-Jan-99	a friend with a special name!
	5	Bryan	Lore	\$1,000.00	\$900.00	24-Dec-01	a brother-in-law
	6	Amy	Lore	\$1,000.00	\$100.00	24-Dec-01	a sister-in-law
	14	Bill	Gates	\$2,000,000,000.00	\$89,992.00	01-Jun-02	
	116	Jane	Doe	\$1,000.00	\$420.00	01-Jan-04	
	866	Bad		\$1,000,000.00	\$235,000.00	01-Apr-04	not a nice person...
*	0				\$0.00		

	OID	CID	DateOfOrder
	9906	3	02-Jan-99
	12351	116	01-Jan-04
	22209	1	15-Jan-04
	22210	1	15-Jan-04
	33410	1	
*	0	0	

	OID	PID	QuantityOrdered	HasShipped
	9906	4	10	<input checked="" type="checkbox"/>
	12351	4	100	<input checked="" type="checkbox"/>
	12351	6	1	<input checked="" type="checkbox"/>
	12351	8	1	<input checked="" type="checkbox"/>
	22209	2	1	<input checked="" type="checkbox"/>
	22210	2	1	<input checked="" type="checkbox"/>
	33410	3	1	
	33410	8	2	
	33410	4	100	
*	0	0		1

	PID	ProductName	QuantityInStock	PricePerItem	Description
	1	Flying Squirrels	3	\$899.99	yes, they really do fly!
	2	Cats	100	\$19.99	
	3	Dogs	20	\$79.03	we carry dalmations only
	4	Ants	10000	\$0.09	
	5	Birds	1000	\$4.95	
	6	Elephants	10	\$389.95	
	7	Racoons	25	\$2.25	
	8	Cobras	25	\$105.00	beth's favorite
*	0		0	\$0.00	

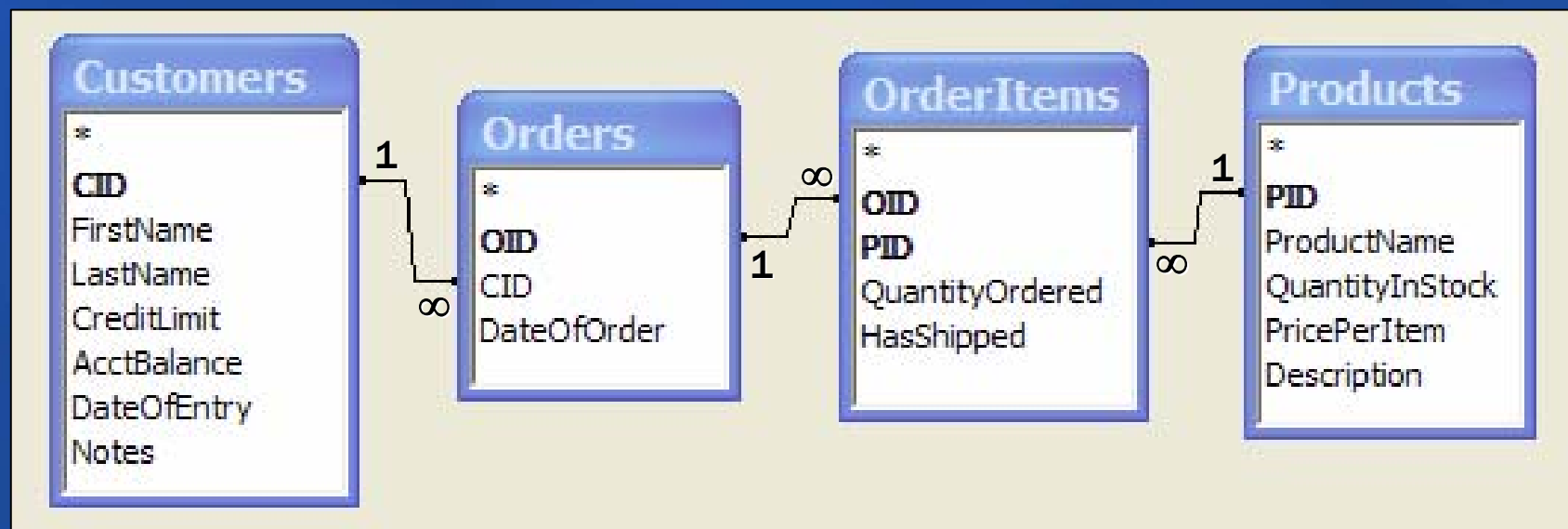




# 今天的数据库模式

您的潜力，我们的动力  
**Microsoft**  
微软(中国)有限公司

- 非常简单的四个表



➤ 注意：对于实际工程来说，这个太简单了，但是对于入门，足够了

# 日程

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

- 一个例子
- ADO.NET 体系结构
- 读取数据
- 写入数据

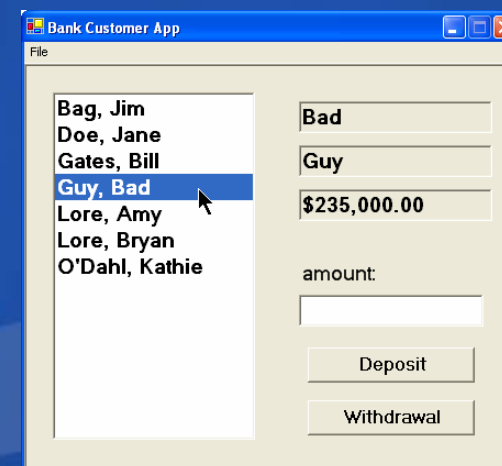
# 例子

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

- 建立简单数据库程序:

1. 打开数据库连接
2. 执行查询语句并取回记录
3. 关闭数据库连接



► 本例采用**MS Access**数据库

# (1) 打开数据库连接

您的潜力, 我们的动力

**Microsoft**  
微软(中国)有限公司

- **ADO.NET Connection 对象:**

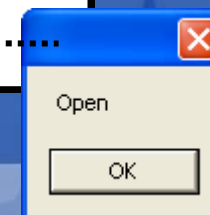
```
using SD=System.Data;           // general DB classes
using MSA=System.Data.OleDb;    // OleDb databases, e.g. MS Access
using SWF=System.Windows.Forms;
:
:

string sConnect;
sConnect = string.Format("Provider={0};Data Source={1}{2}",
    "Provider=Microsoft.Jet.OLEDB.4.0;",
    System.AppDomain.CurrentDomain.BaseDirectory,
    "Sales.mdb");

MSA.OleDbConnection dbConn;
dbConn = new MSA.OleDbConnection(sConnect);
dbConn.Open();

SWF.MessageBox.Show( dbConn.State.ToString() );
```

connection





## (2) 执行查询和取回数据

您的潜力, 我们的动力

**Microsoft**  
微软(中国)有限公司

### ● *Command and Reader* 对象:

```
string sql; BankCustomer c;  
sql = "Select * From Customers Order By LastName Asc, FirstName Asc;"
```

```
MSA.OleDbCommand dbCmd;  
dbCmd = new MSA.OleDbCommand(sql, dbConn);
```

```
MSA.OleDbDataReader dbReader;  
dbReader = dbCmd.ExecuteReader();
```

```
while (dbReader.Read()) // retrieve records from server 1-by-1...  
{  
    c = new BankCustomer(dbReader["FirstName"].ToString(),  
                        dbReader["LastName"].ToString(),  
                        System.Convert.ToDecimal(dbReader["AcctBalance"]));  
  
    Globals.Customers.Add(c); // add this customer to global collection...  
}
```

SQL

reader

record

## (3) 关闭数据库连接

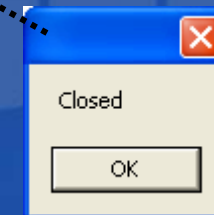
您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

- 关闭 reader 和 connection:

```
dbReader.Close();  
dbConn.Close();
```

```
SWF.MessageBox.Show( dbConn.State.ToString() );
```



您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

# Demonstration One

*demo*

Bank Customers...

The screenshot shows a window titled "Bank Customer App" with a menu bar containing "File". The main area is divided into two sections. On the left is a list box containing the following names: "Bag, Jim", "Doe, Jane", "Gates, Bill", "Guy, Bad" (which is highlighted with a mouse cursor), "Lore, Amy", "Lore, Bryan", and "O'Dahl, Kathie". On the right is a form with three text input fields: the first contains "Bad", the second contains "Guy", and the third contains "\$235,000.00". Below these fields is a label "amount:" followed by an empty text input field. At the bottom right are two buttons: "Deposit" and "Withdrawal".

# 异常处理

您的潜力, 我们的动力

**Microsoft**  
微软(中国)有限公司

- 使用 **try-catch-finally** 来保证收尾工作
  - 数据库关闭, 资源被释放等等

```
OleDbConnection dbConn = null;
OleDbDataReader dbReader = null;

try {
    dbConn = new OleDbConnection(sConnect);
    dbConn.Open();
    :
}
catch(Exception ex) { // see defensive programming webcast...
    :
}
finally {
    try{ dbReader.Close(); } catch{...}
    try{ dbConn.Close(); } catch{...}
}
```

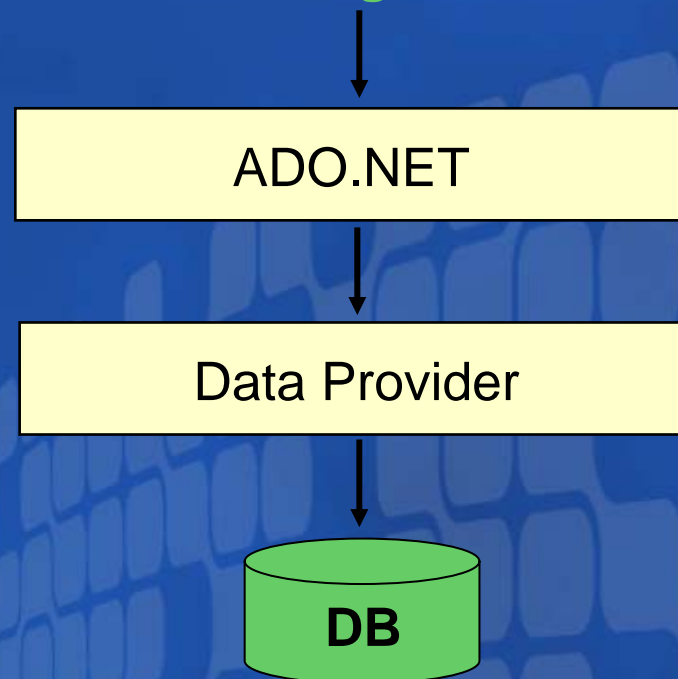
# ADO.NET 体系结构

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

- **ADO.NET 基于Data Provider**
  - Data Provider 封装特定数据库的细节信息

.NET Programmer





# Data Provider

- .NET 可以支持多种数据提供者：
  - Providers for SQL Server and Oracle
  - Support for OLEDB & ODBC
- 其他的 providers :
  - Oracle, MySQL, IBM, ...
    - <http://otn.oracle.com/tech/windows/odpnet/content.html>
    - <http://www.mysql.com/>, <http://crlab.com/mysqlnet/>

# 对象模型

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

- 不同数据库之间
  - “相同”的编程模型，不管使用何种数据库
  - 不同的数据库连接字符串
  - 不同的名称空间和类：
    - *System.Data.SqlClient.SqlConnection*
    - *System.Data.OracleClient.OracleConnection*
    - *System.Data.Odbc.OdbcConnection*
    - etc.

# Demonstration Two

## demo

## Accessing SQL Server...

```
using SD=System.Data;           // general DB classes
using SQL=System.Data.SqlClient; // Microsoft SQL Server
:
:

string sConnect;
sConnect = string.Format("Server={0};Database={1};{2}",
    "localhost",
    "Sales",
    "Integrated Security=SSPI"); // "uid=sa;pwd=???"

SQL.SqlConnection dbConn;
dbConn = new SQL.SqlConnection(sConnect);
dbConn.Open();
```

# Connection 字符串格式

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

- **Connection strings** 对于不同的数据库来说是不同的，格式是数据库提供厂商定义的。现在还没有一个很好的文档包括所有的格式
- 哪里去寻找帮助？
  - <http://www.connectionstrings.com>
  - <http://www.carlprothman.net/Default.aspx?tabid=81>

# Executive 概要

您的潜力, 我们的动力

**Microsoft**  
微软(中国)有限公司

- 几种访问数据库的方式:

任务	ADO.NET
连接的, 读取并返回一个值	<i>ExecuteScalar()</i>
连接的, 只读, 单向	<i>ExecuteReader()</i>
连接的, 只写	<i>ExecuteNonQuery()</i>
非连接的, 读/写	<i>DataSets</i>



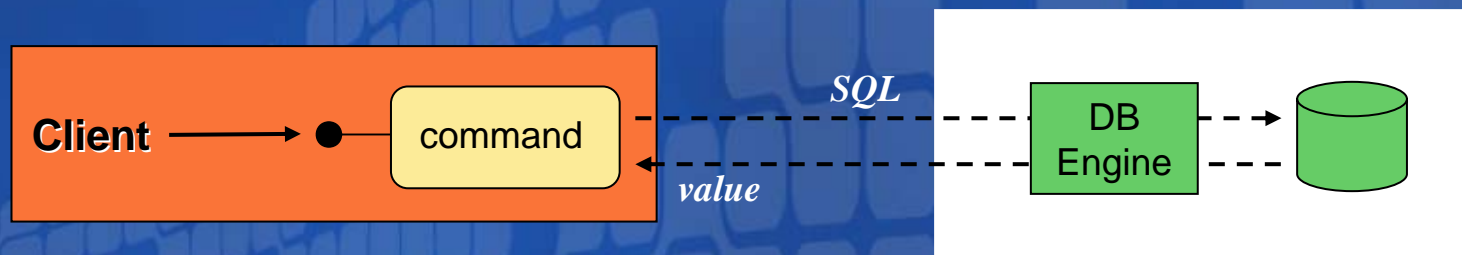
# ExecuteScalar

您的潜力, 我们的动力

**Microsoft**  
微软(中国)有限公司

- 只返回一个值:

```
string sql;  
sql = string.Format("Select AcctBalance From Customers Where CID={0};", cid);  
  
SQL.SqlCommand dbCmd;  
dbCmd = new SQL.SqlCommand(sql, dbConn);  
  
decimal balance;  
balance = System.Convert.ToDecimal( dbCmd.ExecuteScalar() );
```



# ExecuteReader

您的潜力，我们的动力

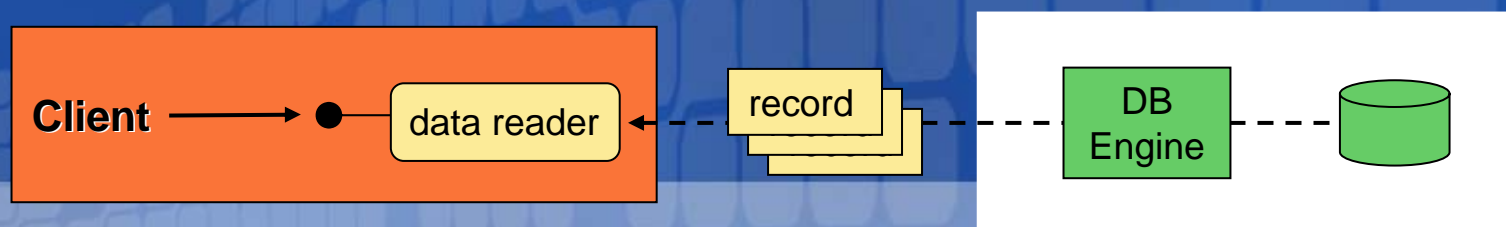
**Microsoft**  
微软(中国)有限公司

- ExecuteReader 产生一个 *DataReader* 对象
- DataReader = forward-only, read-only cursor
  - 没被关闭之前都保持数据库的连接

```
string sql = "Select * From Customers;";

SQL.SqlCommand dbCmd = new SQL.SqlCommand(sql, dbConn);
SQL.SqlDataReader dbReader = dbCmd.ExecuteReader();

while ( dbReader.Read() )
{ ... }
```

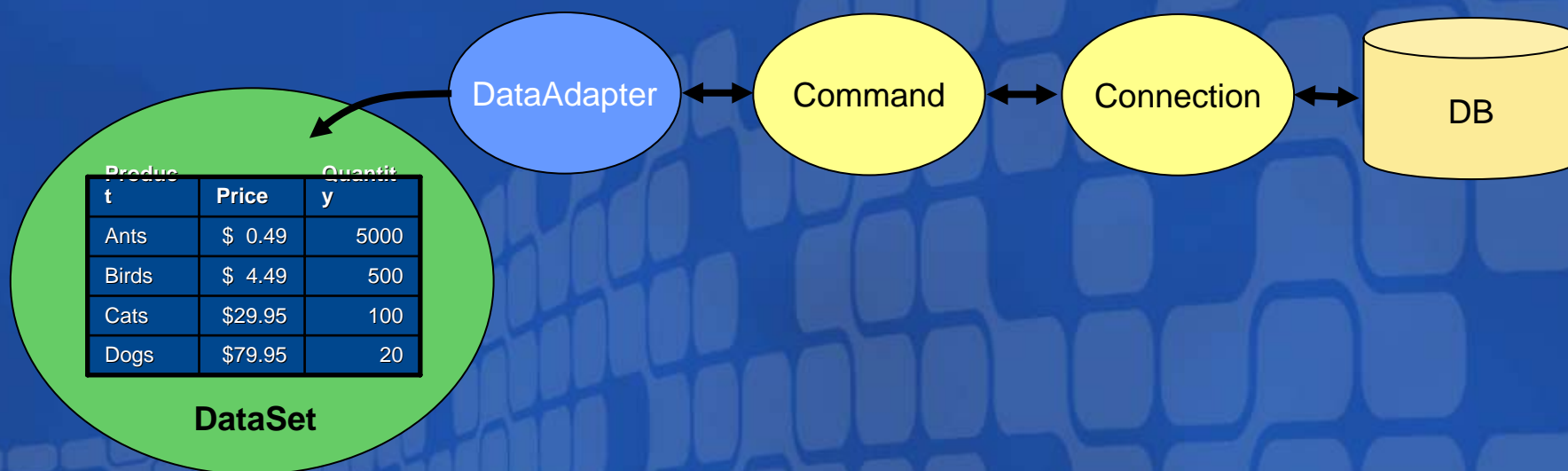


# DataSets

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

- **DataSets** 是另一个 **.NET** 集合类
  - 特别适合象表格形式存储数据
  - 简单填充，传递，显示在**GUI**中



# 为什么使用 DataSet?

您的潜力, 我们的动力

**Microsoft**  
微软(中国)有限公司

- **DataSet = 非连接的记录集**
  - **DataSet** 是服务器端填充为结果的数据拷贝
  - **Dataset** 在客户端运行和处理
  - **DataSet** 可以重新连接数据库用于更新数据
- **好处:**
  - 减少**DB**服务器的负担
  - 关闭连接后离线工作
- *对于分布式程序是非常重要的...*

# 填充 DataSet

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

- 读取并填充DataSet

- 使用DataAdapter 对象

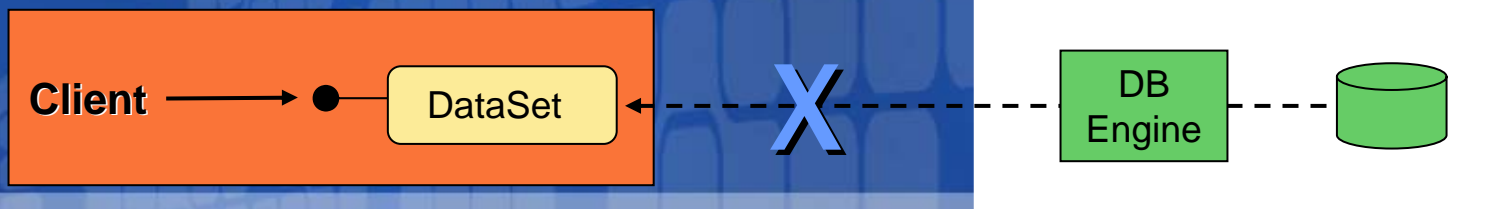
- 当DataSet被填充之后，与数据库服务器的连接就关闭了！

```
string sql = "Select * From Customers;";

SQL.SqlCommand dbCmd = new SQL.SqlCommand(sql, dbConn);
SQL.SqlDataAdapter dbAdapter = new SQL.SqlDataAdapter(dbCmd);

System.Data.DataSet ds;
ds = new System.Data.DataSet();

dbConn.Open();
dbAdapter.Fill(ds);
dbConn.Close();
```





您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

# Demonstration Three

## *demo*

DataSets...

The image shows a screenshot of a Windows application titled "Bank Customer App". The application has a menu bar with "File". The main window is divided into two panes. The left pane contains a list of customer names: "Bag, Jim", "Doe, Jane", "Gates, Bill", "Guy, Bad" (which is highlighted with a blue background and a mouse cursor), "Lore, Amy", "Lore, Bryan", and "O'Dahl, Kathie". The right pane contains three text input fields: the first is labeled "Bad", the second is labeled "Guy", and the third contains the value "\$235,000.00". Below these fields is a label "amount:" followed by an empty text input field. At the bottom of the right pane are two buttons: "Deposit" and "Withdrawal".

# 使用 DataSet

您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

- For each record...

```
string sql = "Select * From Customers;";
:
:
dbConn.Open();
dbAdapter.Fill(ds);
dbConn.Close();

foreach(System.Data.DataRow row in ds.Tables["Table"].Rows)
{
    c = new BankCustomer(row["FirstName"].ToString(),
                        row["LastName"].ToString(),
                        System.Convert.ToDecimal(row["AcctBalance"]));

    Globals.Customers.Add(c); // add to global collection
}
```

The screenshot shows a Windows application titled "Bank Customer App". It features a list box on the left containing the following names: Bag, Jim; Doe, Jane; Gates, Bill; Guy, Bad; Lore, Amy; Lore, Bryan; and O'Dahl, Kathie. The "Guy, Bad" entry is selected and highlighted in blue. To the right of the list box are three text input fields: the first contains "Bad", the second contains "Guy", and the third contains "\$235,000.00". Below these fields is a label "amount:" followed by an empty text input field. At the bottom right are two buttons: "Deposit" and "Withdrawal". A dashed line points from the "Guy, Bad" entry in the list box to the code block on the left.

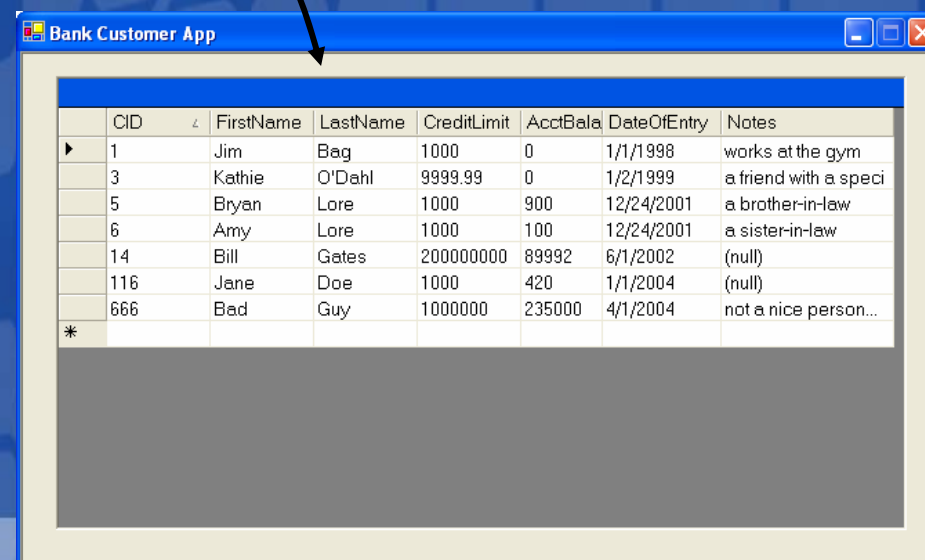
# 绑定到DataGrid

您的潜力，我们的动力

**Microsoft®**  
微软(中国)有限公司

- **DataSet** 可以被绑定到*DataGrid*控件
  - 可以写一行代码完成显示!

```
string sql = "Select * From Customers;";  
:  
:  
this.dataGrid1.SetDataBinding(ds, "Table");
```



	CID	FirstName	LastName	CreditLimit	AcctBala	DateOfEntry	Notes
▶	1	Jim	Bag	1000	0	1/1/1998	works at the gym
	3	Kathie	O'Dahl	9999.99	0	1/2/1999	a friend with a speci
	5	Bryan	Lore	1000	900	12/24/2001	a brother-in-law
	6	Amy	Lore	1000	100	12/24/2001	a sister-in-law
	14	Bill	Gates	200000000	89992	6/1/2002	(null)
	116	Jane	Doe	1000	420	1/1/2004	(null)
	666	Bad	Guy	1000000	235000	4/1/2004	not a nice person...
*							

# DataSets 像数据库

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

- 无限制的表空间
- 可以被查询，排序，更新
- 支持视图，关系，约束

*// fill DataSet with entire Sales database!*

```
string sql = "Select * From Customers; " +  
             "Select * From Orders; " +  
             "Select * From OrderItems; " +  
             "Select * From Products;";  
  
.  
.  
.  
  
dbConn.Open();  
dbAdapter.Fill(ds);  
dbConn.Close();
```

# ExecuteNonQuery

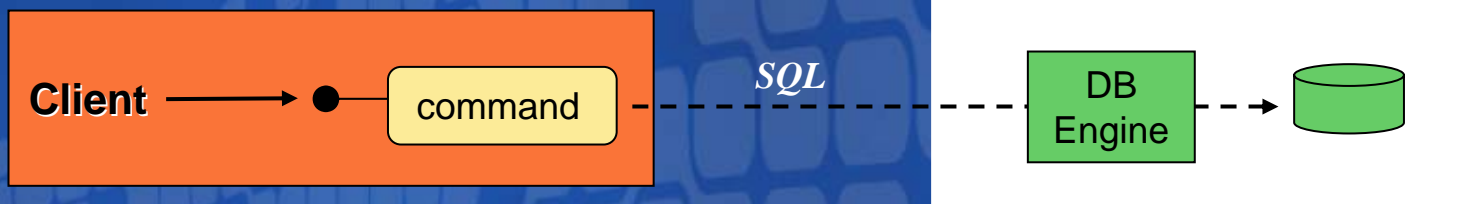
您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

- 写入数据库

- 通过SQL动作：插入，更新和删除来对数据库进行写操作

```
string sql; int rows;  
sql = string.Format("Delete From Customers Where...");  
  
SQL.SqlCommand dbCmd;  
dbCmd = new SQL.SqlCommand(sql, dbConn);  
  
dbConn.Open();  
rows = dbCmd.ExecuteNonQuery();  
dbConn.Close();  
  
if (rows != 1) throw new System.ApplicationException("Delete failed?!");
```





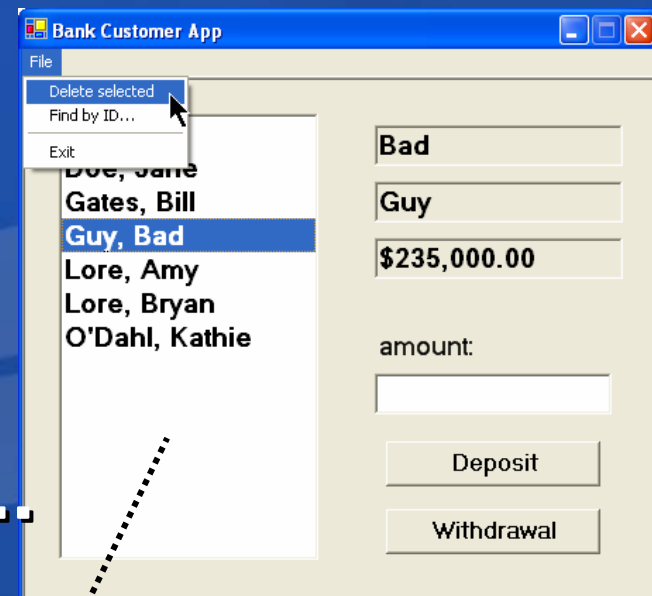
您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

# Demonstration Four

## *demo*

## Deleting by text fields...



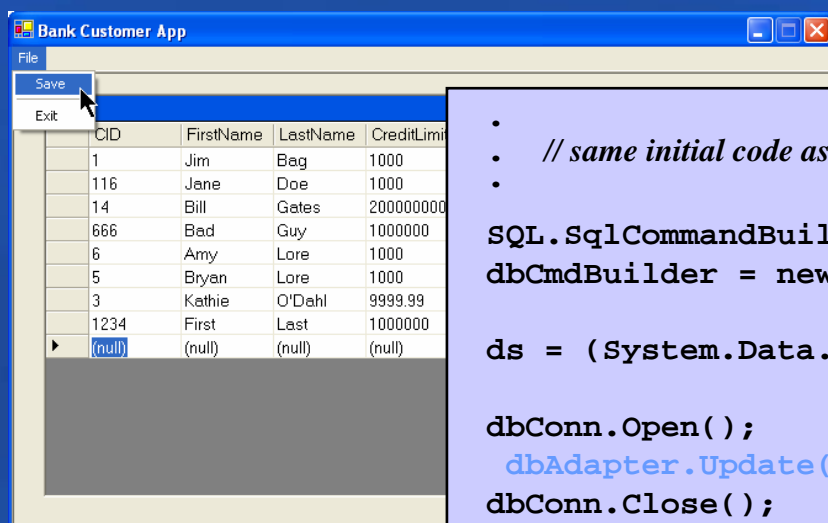
```
BankCustomer bc = (BankCustomer) this.lstCustomers.SelectedItem;  
  
sql = string.Format("Delete From Customers Where FirstName='{0}' and LastName='{1}';",  
                    bc.FirstName.Replace("'", "''"), bc.LastName.Replace("'", "''"));
```

# DataSets — 写入数据库...

您的潜力，我们的动力

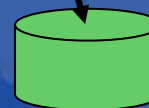
**Microsoft**  
微软(中国)有限公司

- **DataSets 允许非连接的修改**
  - 客户端应用程序重新连接后更新



```
.  
.  
// same initial code as when filling dataset  
.  
  
SQL.SqlCommandBuilder dbCmdBuilder;  
dbCmdBuilder = new SQL.SqlCommandBuilder(dbAdapter);  
  
ds = (System.Data.DataSet) this.dataGrid1.DataSource;  
  
dbConn.Open();  
dbAdapter.Update(ds);  
dbConn.Close();
```

- 小心: 非连接容易引起很多的冲突...



# Session Summary

您的潜力, 我们的动力

**Microsoft**  
微软(中国)有限公司

- 使用VS的工具来创建数据库连接:
  - Example: Project menu, Add New Item..., Data Form Wizard
- 否则, 还有一些方法访问数据库:
  - Need to read a single value? Prefer *ExecuteScaler*
  - Need to modify a database? Prefer *ExecuteNonQuery*
  - Need to read a large # of records? Prefer *ExecuteReader*
  - Need to read & pass data around? Prefer a *DataSet*
- 今天我们讲述的只是冰山一角...

# 补充：参数化查询

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

- 参数化的形式

```
Select FirstName, LastName  
From Customers  
Where CID = @CID
```

```
sql = "Select * From Customers Where CID=@CID Order By LastName Asc,  
FirstName Asc;";
```

```
SQL.SqlCommand dbCmd;
```

```
dbCmd = new SQL.SqlCommand(sql, dbConn);
```

```
dbCmd.Parameters.Add("@CID", SqlDbType.BigInt);
```

```
dbCmd.Parameters[0].Value = 1;
```

```
dbReader = dbCmd.ExecuteReader();
```

# 补充：调用存储过程

您的潜力，我们的动力

**Microsoft**  
微软(中国)有限公司

- **SQL Server中的存储过程**

```
Create PROCEDURE GetCustomer (@CID BigInt) AS  
    Select FirstName, LastName, AcctBalance From Customers Where CID = @CID  
RETURN
```

```
SQL.SqlCommand dbCmd = dbConn.CreateCommand();  
  
dbCmd.CommandText = "GetCustomer";  
  
dbCmd.CommandType = CommandType.StoredProcedure;  
  
dbCmd.Parameters.Add("@CID", SqlDbType.BigInt);  
  
dbCmd.Parameters[0].Value = 1;  
  
dbReader = dbCmd.ExecuteReader();
```



# 补充: Transaction中执行查询

您的潜力, 我们的动力

**Microsoft**  
微软(中国)有限公司

- **Command对象的BeginTransaction来创建一个Transaction对象**

```
SQL.SqlTransaction txn = dbConn.BeginTransaction();

string strSQL = "INSERT INTO Customers
VALUES(100,'Hui','Yu',1000000,1000000,'01-Jan-2005','money')";

SQL.SqlCommand dbCmd;

dbCmd = new SQL.SqlCommand(strSQL, dbConn,txn);

int intRecord = dbCmd.ExecuteNonQuery();

if(intRecord ==1){SWF.MessageBox.Show( "Update
Successful");txn.Commit();}

else          {SWF.MessageBox.Show( "Update Failed"); txn.Rollback();}
```

# 其它参考资源

您的潜力, 我们的动力

**Microsoft**  
微软(中国)有限公司

- **Webcast:**

- <http://www.microsoft.com/china/msdn/events/webcasts/shared/Webcast/MSDNWebCast.aspx>

- **Resources:**

- MSDN Library  
<http://www.microsoft.com/china/MSDN/library/default.mspix>
- **David Sceppa “ADO.NET CORE REFERENCE”**  
( ADO.NET技术内幕, 清华)

**MSDN flash:**

- <http://www.microsoft.com/china/newsletter/case/MSDN.asp>

# 感谢大家的参与!

您的潜力, 我们的动力

**Microsoft**  
微软(中国)有限公司

- 非常感谢大家的参与
- 您的**反馈**对我们来说是非常重要的




Date	Topic
2005年8月1日	C#中WinForms应用
2005年8月16日	ADO.NET
2005年9月	多层应用程序开发




# Q&A

如需提出问题，请单击“提问”按钮并在随后显示的浮动面板中输入问题内容。一旦完成问题输入后，请单击“提问”按钮。

 **问题和解答 (无问题)** ▲ ×

在此会议中尚未解答任何问题。

要向演示者提问，请在此处键入问 

提问(A)

删除(D)

问题管理器(Q)

您的潜力，我们的动力

**Microsoft®**  
微软(中国)有限公司

**Microsoft®**

msdn  


**MSDN Webcasts**