

12.1以基础类的数据源提供程序模型 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/140/2021_2022_121_E4_BB_A5_E5_9F_BA_c100_140286.htm 2005年底（2005年10月）与 SQL Server 2005一起出现的是 .NET Framework 2.0 版本，其中用来访问数据库的 ADO.NET 类也升级到 ADO.NET 2.0 版。ADO.NET 2.0 除了增强旧功能外，也提供了相当多的新功能，包含了以基础类为本（base-class-based）的数据源提供程序（provider）模型、异步访问架构、批处理更新与大量数据复制（bulk copy）、SQL Server 2005 的回调通知、单一连接同时多执行结果集（MARS）、执行统计、强化的 DataSet 类等等。换句话说，若要有效发挥 SQL Server 2005 的功能，前端应用程序最好用 ADO.NET 2.0 来开发。ADO.NET 2.0 提供了相当多的新增功能，一些与数据源提供程序无关，也就是访问各种数据库都可以用到的功能，但有很大的一部分是专属于 SQL Server 2005，针对 SQL Server 2005 的新功能提供给前端应用程序开发使用。以下针对各种新功能作一个概观性的介绍：12.1 以基础类为本的数据源提供程序模型 ADO.NET 提供的多种类可用来访问各个不同的数据源，如 SQL Server、Access/Jet、Oracle、NCR 的 Teradata 等等，由于数据访问的操作同质性很高，我们会期待以相同的方式操作不同的数据源，其内建置相同或近似的数据库结构。在 ADO.NET 1.0 与 1.1 版本时为各种数据源提供了一系列不同的 Provider，如 SqlClient、OracleClient、OLEDB、ODBC 等等。广泛性一致访问（generic）的程序编写只能靠制定标准公共接口（generic interface）来完成。例如 System.Data.SqlClient 命名空间内的

SqlConnection 类和 System.Data. OracleClient 的 OracleConnection 类都实现了 IDbConnection 接口。在新的 ADO.NET 2.0 则以一系列的基类 (Base Class) 为主, 这些类所属的命名空间是 System.Data.Common, 以此提供相同的功能, 这称为 “ 以基础类为主的数据源提供程序模型 (Base-Class-Based Provider Model) ”。但为了向前兼容, ADO.NET 2.0 依然实现了以往的通用界面, 让其他的 Provider 开发者可以选择要衍生基础类或是实现相同界面的方式来编写数据提供程序 (Provider)。在此, 我们提供一个简单的范例程序解说基础类的使用方式。首先你可以查看一下 Access (Jet) 提供的范例数据库 Northwind.mdb (默认安装目录放在 C:\Program Files\Microsoft Office\OFFICE11\SAMPLES) , 并以 Access 自身提供的 “ 工具 ” - “ 数据库实用工具 ” - “ 升迁向导 ” , 在 SQL Server 内创建一个一模一样的数据库。我们就以相同的程序代码让你在程序执行时选择要访问的数据库 (Jet 或 SQL Server)。接着, 在 Visual Studio 打开 Visual Basic.NET 的 Windows 应用程序项目, 而范例程序的设计与执行画面如图12-1所示, 你可以通过左上方的下拉选单选择不同的数据源: 图12-1 通过相同的程序代码访问不同类型的数据库在 app.config 文件中的 appSettings 元素内加入以下四行设置, 分别设置不同 .NET 数据提供程序的类型名称, 以及该数据提供程序的连接属性, 我们在程序代码中将会读取这些属性以创建正确的数据提供程序, 并通过该提供程序连接到数据源: 简单的程序代码范例如列表12-1: 程序代码列表12-1 通过相同的程序代码访问不同的数据库产品为不同的 DataProvider 创建 ProviderFactory 此处有两种选择

System.Data.SqlClient 和 System.Data.OleDb 大小写有别

```
Dim myFactory As DbProviderFactory =  
_DbProviderFactories.GetFactory  
( ConfigurationSettings.AppSettings ( ComboBox1.Text ) ) 通  
过 Factory 创建 Connection Using cnn As DbConnection =  
myFactory.CreateConnection ( ) 取回存放在 app.config 内的连  
接属性设置 cnn.ConnectionString = _  
ConfigurationSettings.AppSettings ( ComboBox1.Text & amp.  
"Connection" ) cnn.Open ( ) 通过 DbConnection 创建个别的  
DbCommand Using cmd As DbCommand = cnn.CreateCommand  
( ) cmd.CommandText = _ "SELECT * FROM 客户 WHERE 客  
户编号 LIKE @CustomerID" 通过 Factory 创建 DbParameter 相  
同的参数创建方式 , 设置不同数据提供者的参数内容 Dim  
param As DbParameter = myFactory.CreateParameter ( )  
param.ParameterName = "@CustomerID" param.Value =  
txtCustomerID.Text cmd.Parameters.Add ( param ) 取回数据结  
果后 , 通过 DataBinding 让 DataGridView 显示数据表的内容  
Dim tbl As New DataTable tbl.Load ( cmd.ExecuteReader ( ) )  
DataGridView1.DataSource = tbl End Using End Using 100Test 下  
载频道开通 , 各类考试题目直接下载。详细请访问  
www.100test.com
```