

11.2 Visual Studio 2005 内的 SQL Server 方案 Microsoft 认证考试 PDF 转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/562/2021\\_2022\\_112Visual\\_c100\\_562955.htm](https://www.100test.com/kao_ti2020/562/2021_2022_112Visual_c100_562955.htm) 首先查看一下 ADO.NET 2.0 提供用来创建 SQL

Server 服务器端对象的相关类。11.2.1 System.Data.SqlServer 命名空间当用 .NET 开发 SQL Server 的内部对象时，最常使用的命名空间将会是存在 Sqlaccess.dll 程序文件内的

System.Data.SqlServer。若你曾经用 ADO.NET 开发数据库的应用程序，则部分对象是相通的。例如 SqlConnection

、SqlCommand 等，微软的开发小组尽量让程序设计师在 SQL Server 内访问数据与在外部访问数据的程序编写方式相似，而不需要熟悉两套开发模式。以下稍微解释这些对象

：SqlContext：SqlContext 类让你可以取得 SQL Server 内部调用者的执行环境(Context)，主要是取得 SqlPipe 类实例，用此返回结果给前端调用该数据库对象的应用程序，或是

SqlTriggerContext 以提供编写触发器 (Trigger) 的相关数据

。SqlContext 类在让我们所编写的程序集与 SQL Server 进行沟通时显得非常重要。使用该类的程序范例如程序代码列表

11-1 所示：程序代码列表 11-1 通过 SqlContext 取得用户当前的执行环境

```
Using cnn As New SqlConnection("Context Connection=true") Using sqlCmd As New SqlCommand 因为 Product 数据表属于不同的 Schema，所以需要加上 Schema 名称
```

```
cnn.Open() sqlCmd.Connection = cnn sqlCmd.CommandText = "SELECT ProductNumber FROM Production.Product" Dim sqlRead As SqlDataReader = sqlCmd.ExecuteReader() 因为是存储过程，要通过 SqlPipe 将运行结果接回原来 SQL Server 的输出
```

```
Dim sqlP As SqlPipe = SqlConnection.Pipe sqlP.Send(sqlRead) End
UsingEnd Using
```

程序代码列表 11-1 仅是简单地取得 Production.Product 数据表内的 ProductNumber 字段，然后直接返回给前端。由于被调用的对象是在用户已经取得的连接环境内执行，若要传递数据给用户，就需要通过程序代码列表 11-1 的方式引用 SqlConnection 类的 Pipe 属性，取得 SqlPipe 对象实例后，再通过 Send 方法将结果返回。

**SqlConnection**：让我们所编写的程序取得对程序集实例被 SQL Server 外部的前端应用程序调用时，该应用程序所在的数据库连接相关设置。这些连接设置并不是由我们用 .NET 所编写的存储过程或用户自定义函数的对象决定的，被调用的数据库内部对象需要与外部应用程序使用相同的连接环境。可以用以下的程序通过连接字符串设置如下，而通过这条连接再执行其他的 SQL 语法：

```
Using cnn As New SqlConnection("Context
Connection=true")
```

**SqlCommand**：让你传送 T-SQL 命令到数据库服务器，参照前述程序代码列表 11-1 的范例，可以依照一般方式创建 SqlCommand 对象实例，并通过 SqlConnection 实例来执行取得产品编号的 T-SQL 语法。

**SqlParameter**：用以设置与访问 SqlCommand 对象内传递的参数。范例程序如程序代码列表 11-2 所示：

程序代码列表 11-2 通过 SqlCommand 和 SqlParameter 对象搭配 SHA1 哈希算法为密码编码

```
Public Shared Sub WriteHashedPassword(ByVal UserName As String,
ByVal Password As String)
将账号/密码经过 Sha1 Hash 算法换算后，再存入数据表
Try Using cnn As New
SqlConnection("Context Connection=true") cnn.Open() Using
sqlCmd As New SqlCommand Dim sh1 As New SHA1Managed
```

```
Dim uEncode As New UnicodeEncoding 将密码以 SHA1 计算哈希值，再以 Base64 编码。 Dim txtBytes As Byte() =
uEncode.GetBytes>Password) Dim hashedPassword As Byte() =
sh1.ComputeHash(txtBytes) Dim strHash As String =
Convert.ToBase64String(hashedPassword) Dim dml As String =
"INSERT NamePass VALUES(@UserName,@HashAsString)" With
sqlCmd .Connection = cnn .CommandText = dml 通过
SqlParameter 对象设置 SqlCommand 对象所包含 T-SQL 语法 需
要的参数 .Parameters.Add(New SqlParameter("@UserName",
SqlDbType.NVarChar, 50)) .Parameters(0).Value = UserName
.Parameters.Add(New SqlParameter("@HashAsString",
SqlDbType.NVarChar, 100)) .Parameters(1).Value = strHash
sqlCmd.ExecuteNonQuery() End With End Using cnn.Close() End
UsingCatch ex As Exception Dim fs As New
FileStream("C:\YukonCLR.log", FileMode.OpenOrCreate,
FileAccess.Write) Dim sw As New StreamWriter(fs)
sw.WriteLine(ex.ToString()) sw.Close()End TryEnd Sub 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
```