

VisualBasic中DataCombo和DataList控件的使用 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/137/2021_2022_VisualBasi_c97_137523.htm 使用 DataCombo 和 DataList 控件 DataCombo 和 DataList 控件与标准列表框和组合框控件极为相似，但有一些重要的不同之处，这种不同使这两个控件在数据库应用程序中具有极大的适应性和用武之地。这两个控件都可以被这些控件所绑定的数据库字段自动填充。此外，它们还能有选择地将一个选定的字段传递给第二个数据控件，从而适合用于创建“查找表”应用程序。可能的用法

1. 在一个关系数据库中，使用一个表的数据来提供要输入给第二个（相关的）表值。例如，在一个存货清单数据库中，供应商的名称存储在一个表中，每个供应商都有一个唯一的标识符。另一个显示产品的表则使用这些标识符来表明是哪个供应商供应的该产品。可以使用 DataList 控件来显示供应商的名称，而（不可见地）将供应商的标识符提供给产品表。
2. 允许用户通过从一个下拉列表中选择一种标准来缩小搜索范围。例如，一个销售报告的数据库应用程序可以使用一个 DataList 控件让用户选择一个州 (State) 或一个销售区域。一旦作出选择，则该选择项将自动传递给第二个数据控件，这个控件负责查找选定区域的销售记录。与它们对应的内在控件一样，DataList 和 DataCombo 控件之间的主要不同在于 DataCombo 控件提供了一个可以在其中编辑内容的文本框。

详细信息 有关 DataList 和 DataCombo 控件连接数据库表能力的说明，请参阅“使用 DataCombo 和 DataList 控件连接两个表”。要创建一个简单的使用链接表的数据库应用程序，请参阅“创建一个简单的

DataCombo 应用程序”。值得注意的控件属性 DataList 和 DataCombo 控件的一些重要属性包括：注意 DataCombo 控件的 DataFormat 属性是一个 Extender 属性。因此在属性表上它总是可见的，并且可以在代码中设置。然而，DataCombo 控件仅对其列表中最上端的项格式化。对于看到已格式化的顶端项的最终用户来说，这一点可能不太重要，只要从未格式化的列表中选择即可。已格式化的项可能也会误导最终用户，使他们以为项目要在格式化之后再输入数据库。由于这些原因，建议在使用 DataCombo 控件时不要设置 DataFormat 属性。详细信息如果要使用一个演示 BoundText 属性用法的循序渐进教程，请参阅“创建一个连接 DataList 控件的 DataGrid”。关于这些控件的属性和方法的完整列表，请参阅“DataList 控件”和“DataCombo 控件”。

使用 DataCombo 和 DataList 控件连接两个表 DataCombo 和 DataList 控件与众不同的特性是具有访问两个不同的表，并且将第一个表的数据链接到第二个表的某个字段的能力。这是通过使用两个数据源完成的（诸如 ADO Data 控件或 Data 环境）。关系表和“不友好的”值在一个关系数据库中，对于重复使用的信息并不是在多个地方都保存其全部的信息。大多数这种信息都保存在由多个字段组成的一个记录集中；在这些字段中有一个“标识符”字段来唯一地标识这个记录集。例如，VisualBasic 提供的 Biblio 数据库在一个名为 "Publishers" 的表中存储了若干个出版公司的名称。这个表包括很多字段，诸如地址、城市、邮政编码以及电话号码等。但是为了简单起见，只考虑这个表的两个本质字段 Name 和 PubID 字段。Name 字段存储一个出版商的名称，而 PubID 字段则存储一个相对“不友好

的”值，如一个数或代码。但这个不友好的值是很重要的，因为这个值唯一地标识该出版商，并且可以作为一种链接整个记录集的手段。此外，这个值会存储在第二个表中的多个记录集中。第二个表的名称为"Titles"，其每个记录集包含的信息包括标题、出版年份、国际标准书号 ISBN 等。在这些字段中有一个字段的名称就是"PubID"。这个字段的名称与 Publishers 表中的相应字段的名称相同，因为这个字段存储了将该标题和一个特定的出版商链接在一起的值。这种可行方案提出了一个小问题：给定一个允许用户插入新标题的数据库应用程序，用户必须用某种方法输入标识出版商的整数。如果用户能记住每个出版商的唯一标识符，那么也还是可行的，不过如果一方面用户能看到出版商的名称，另一方面存入应用程序的又是数据库中相应的值，则会显得更加方便。而 DataList 和 DataCombo 控件就可以轻松地解决这个问题。两个数据源、三个字段、无编码 DataList 和 DataCombo 控件使用两个数据源来解决这个问题。在只显示出版商的名称（来自 Publishers 表）的同时，DataList 或 DataCombo 控件只将 PubID 字段的值写入到 Titles 表。通过“属性”窗口，将 RowSource 设置为提供要写入的数据的数据源（即 Publishers 表）。然后将 DataSource 属性设置为要写入数据的数据源（即 Titles 表）。最后，设置 DataField、ListField 以及 BoundColumn 属性。下图演示了如何将两个数据源（以两个 Data 控件的形式）以及三个字段指定给一个 DataCombo 控件：简要而言，ListField 属性决定该控件所显示的是哪一个字段。在本例中就是出版商的名称。另一方面，BoundColumn 属性则决定 Publishers 表中由哪一个字段向

Title 表供应实际所需的值。注意 Publishers 表中的 PubID 字段不能（也不应该）被编辑。相反，在 PubID 字段中的值将写入到由 DataField 属性所指定的字段。在本例中，这个属性就是 Titles 表中的 PubID 字段。下表概要地介绍这些属性及其使用方法。注意 DataList 和 DataCombo 控件也可以与单个数据控件一起使用。要实现这一点，可以将 DataSource 和 RowSource 属性设置为同一个数据控件，并且将 DataField 和 BoundColumn 属性设置为该数据控件的记录集中的同一个字段。在这种情形下，将使用 ListField 的值来填充该列表，且这些值来自于被更新的同一个记录集。如果指定了一个 ListField 属性，但没有设置 BoundColumn 属性，则 BoundColumn 将自动被设置为 ListField 字段。详细信息如果想使用 DataCombo 控件循序渐进地创建一个简单的数据库应用程序，请参阅“创建一个简单的 DataCombo 应用程序”。

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