

VisualBasicCOM基础之更多属性 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/252/2021\\_2022\\_VisualBasi\\_c97\\_252658.htm](https://www.100test.com/kao_ti2020/252/2021_2022_VisualBasi_c97_252658.htm)

有时候，为了更好地处理类往往需更多的属性。例如，假如你有四种不同的客户群：集团、较大、较小和新的客户类型，或者一个用于搜索的类中有三种不同的搜索方式：软盘、硬盘和网络。那么，能不能最好从选项列表中选择一个，而不是用不能理解的数字或文本来设置相关属性？我想，答案是肯定的。因为这种处理就称为"枚举"。打开上一节的工程，让我们添加一些代码。在CDog类中添加下列代码：  
`Public Enum CoatType  
BigAndShaggy = 1  
ShortCrewCut = 2  
PoodleStyleAfro = 3  
Unknown = 4  
end Enum` 关键词"Enum"就是用来定义枚举的，换句话说，它是可能选项的列表。各选项都有相应的数字，也就是说BigAndShaggy表示1

，ShortCrewCut等于2，等等。需要注意的是，当将枚举项相关信息添加在数据库中时，其相应的数值是非常有用的。由于"BigAndShaggy"实际代表的是数值1，所以可以直接将其插入到数据库的数值字段中。这就意味着，可以方便使用字符串来维护数据库。所以，我们来创建一个Dog的Coat类型列表，并另外定义一个属性，将这些类型添加在CDog类中。在类中声明下列变量：  
`Private udtCoat As CoatType` 这个定义的私有变量用来保存即将添加的Coat类型属性，注意到udtCoat变量既不是字符串也不是整型，而是我们自己定义的枚举类型CoatType。当类CDog打开时，选择"Tools"菜单中的"Add Procedure"命令，弹出相应的对话框；Name编辑框中键入Coat；选中"Property"选项按钮，然后单击[OK]。系统自动

产生下列代码框架：  
`Public Property Get Coat() As Variant End  
Property Public Property Let Coat(ByVal vNewValue As Variant)  
End Property` 但我们需要的却不是这个框架。代码中  
，"Variant"变量类型是能接收和处理任何类型数据。在我们定义的CDog类中，最后的属性是Age，它只能接受整型。但现在需要属性能接收CoatType列表中的数据类型，因此需要作下列修改：  
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](http://www.100test.com)