

如何将界面代码和功能代码分离（基于Delphi_VCL）计算机等级考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/643/2021_2022__E5_A6_82_E4_BD_95_E5_B0_86_E7_c97_643726.htm 很多朋友看了上次我

写的“创建良好设计的代码（基于Delphi/VCL）”后，对我说感觉上可以接受其中的观点，但似乎说得太简单，不够具体；也有的朋友对其中的一个小例子有些异议。因此便有了此文。上次，我举的例子是这样的：假设要从某处获得一个字符串列表，然后显示于 TListBox 中，我所推崇的代码是：

```
ObjectXXX := TObjectXXX.Create. ListBox1.Items :=
```

```
ObjectXXX.GetStringList. ObjectXXX.Free.
```

的确，我承认，单纯从这三行代码来看，似乎有了“滥用对象”之嫌。也许是例子过于简单，给人的感觉是TObjectXXX只有GetStringList这一个public成员函数，如果真的这样的话，那可真是“滥用对象”了。类是对对象的抽象，而对象是由状态和操作（也就是数据和对数据的操作）的集合组成。因此，没有状态的对象不是对象！没有私有数据成员的类的设计是失败的设计（那不是类，而是接口了）。好，下面我就举一个详细的例子来说明，如何将界面代码和功能代码分离。假设我要做一个简单的个人通讯录管理软件，很显然，整个软件分为两部分：一部分是面向用户的，也就是所谓界面部分，我可以提供四个按钮（分别为“添加”、“删除”、“修改”、“查找”）和一个编辑框（显示通讯录信息和接受用户输入）用于和用户交互；另一部分是功能化的，也就是软件内部的对于通讯录的存取操作。于是，有了一个TAddrBook类，它是对功能化部分的抽象。 TAddrBook = class private //一些私有成员

public constructor Create. destructor Destroy.override. GetCount: Integer. FindRecord(strString): Integer. GetRecord(nIndex:Integer): String. SetRecord(nIndex:integer. strRec:String): Boolean. AddRecord(strRec:String): Boolean. DelRecord(nIndex): Boolean. //其它共有成员函数 end. 私有成员之所以无法确定，主要是取决于这个类的实现。如此，可以将对通讯录的存取操作的逻辑封装。而界面部分的代码不会涉及到这些存取逻辑。界面部分代码如下：

```
var Form1: TForm1. AddrBook: TAddrBook. nCurRec: Integer. implementation procedure TForm1.FormCreate(Sender: TObject). begin AddrBook := TAddrBook.Create. nCurRec := AddrBook.GetCount. end. procedure TForm1.FormClose(Sender: TObject. var Action: TCloseAction). begin AddrBook.Free. end. //添加按钮 procedure TForm1.Button1Click(Sender: TObject). begin if not AddrBook.AddRecord(memo1.Text) then ShowMessage(\"error\"). end. //删除按钮 procedure TForm1.Button2Click(Sender: TObject). begin if not AddrBook.DelRecord(nCurRec) then ShowMessage(\"error\"). end. //修改按钮 procedure TForm1.Button3Click(Sender: TObject). begin if not AddrBook.SetRecord(nCurRec, memo1.Text) then ShowMessage(\"error\"). end. //查找按钮 procedure TForm1.Button4Click(Sender: TObject). begin memo1.Text := AddrBook.GetRecord(AddrBook.FindRecord(memo1.Text)). end.
```

以上界面部分的代码，不涉及任何存取逻辑，每个模块的代码简单，易懂，便于维护。而实际上，该通讯录是使用数据库保存还是用文本文件来保存，界面代码都不知道；使用数

数据库的话，是通过ODBC还是ADO还是BDE访问数据库，界面代码也不知道。实际上，这些存取逻辑的东西取决于TAddrBook类的实现，TAddrBook类的实现可以单独的放在一个.pas文件中，对TAddrBook类的实现的任何更改，都不会影响界面部分。维护代码的时候，将更改局限于某一个模块中的做法是非常明智的。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