

再谈多态:多态的应用举例计算机等级考试 PDF转换可能丢失
图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/645/2021_2022__E5_86_8D_E8_B0_88_E5_A4_9A_E6_c97_645018.htm 再谈多态--多态的应用
举例：这一段时间关于多态的讨论可谓一浪高过一浪，各位大虾都有高论出炉。本人也按奈不住，所以结合一下本人的工作实际，谈谈多态的应用问题。在我做系统的早期所应用的多态情况较少,多是出现这样一些用到多态的情况 var s : TStrings. begin s := TStringList.Create. try s.Add(\hello!\). finally s.free. end. end. 上面这段程序可能在很多人的源程序中都出现过,而且以\"TStringList.add\"来搜delphi的help可以看到类似的例子。对于 TStrings由于其方法create是abstract所以不能直接使用必须使用其子类的Create方法。这里就是把子类 (TStringList)的对象实例赋给定义的父类(TStrings)对象.而Add在TStrings中是被声明为virtual的，那么根据多态的特性s对象在使用Add方法时会调用其创建时所使用的类型(TStringList)的方法而不是TStrings。 www. Examda.CoM考试就到百考试题
这里还是体现不出多态的好处，那么就拿我做过的一个表格系统为例子吧。做这个系统的时候要求做一系列表格，基本上是一个表格一个Form 如:TgridForm1,TgridForm2...，然后这些Form内嵌在外部的一个TContainerForm中,由 TContainerForm调用所有表格Form的方法如存盘(save).到这里都没有问题，很简单。注意现在问题来了。系统要求每一个表格Form 独立打包，而且程序开发过后会不断的加入新的未知的表格Form，但不允许将整个程序重新codeing和compile，对于最终用户来说只要down 一个新的包含这个表格的包就能

使用新的表格。这时多态的强大作用就发挥出来了。我是这么做的，做一个TParentgridForm一个所有表格Form的父类，它定义了所有可能的方法(如：save)，当然是Virtual的。这时各个表格Form根据业务要求自行实现Save等方法。对于TContainerForm来说它只知道TParentgridForm不知道具体的表格Form但是由于多态的作用，当TContainerForm在调用内嵌在它里面的Form的Save方法时会正确的调用各个表格Form的Save方法，这样各个表格的业务逻辑就实现了。而且以后不管加多少个新的表格，TContainerForm都会实现其正确功能，并且无需重新Compile整个系统了。哈哈，有点对象插件的味道。采集者退散可见如果大家善于利用多态的话，在现实的编程中会享受到其极大的好处。特别是对于上面这种灵活度要求很高的系统。哈哈讲了这么多不知道大家明白不。

编辑特别推荐: 全国计算机等级考试资料下载 全国计算机等级考试论坛 如何应付二级考试中的公共基础知识 全国计算机等级考试上机考试应试技巧 百考试题教育全国计算机等级考试在线测试平台 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