

VB中访问API函数之防错技巧 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_VB_E4_B8_AD_E8_AE_BF_E9_97_c97_252594.htm 功能强大的API（应用程序接口）函数对于VB（Visual Basic For Windows）程序员来说，不愧是很好的编程工具，然而使用API函数的程序员也许都遇到这样的现象。在VB集成环境下，程序运行后，出现一错误信息对话框，按确定键后系统自动退出VB集成环境，此时如果你的程序尚未存盘，那末很遗憾挽回损失已回天乏力。最后一次存盘之后的程序都不复存在。这是你对API函数使用不当引起的一般保护故障（GPF）。当一个GPF错发生时，你应允许Windows关闭你的应用。有些情况下你可能需要退出Windows或者重新引导系统。出错程度视哪里内存被破坏而定。DLL（动态链接库）函数中的类型不一致等错误是GPF错的主要原因。这些错误会导致GPF甚至使Windows系统完全崩溃（需要重新引导系统）。下面谈避免GPF的一些技巧。用别名来提供强类型检查是避免GPF的有效措施之一。有些情况下，DLL函数可以接受多种类型，LoadCursor函数就是这样一个例子，其定义如下：HCURSOR LoadCursor（hInstance，lpCursorName）这里HCURSOR是一个指向光标对象的16位句柄，hInstance是一个16位实例句柄，lpCursorName是光标的名字或者是光标资源的32位整数ID。为了支持两种类型的lpCursorName参数。VB有必要包含如下两个声明：Declare Function LoadCursorLib " USER "（ByVal hInstance As Integer，ByVal lpCursorName As String）As Integer和Declare Function LoadCursor Lib " USER "（Byval hInstance As Integer，ByVal

lpCursorName As Long) As Integer 但是，这两个声明不能在一个程序中同时存在，因为Visual Basic会报重复声明错。我们知道，As Any声明可使得任何参数都可以传递给DLL函数，因此可以如下声明：`Declare Function LoadCursor Lib " USER " (ByVal hInstance As Integer , ByVal lpCursorName As Any) As Integer` 上述声明意味着Visual Basic能支持一个参数可接受多种类型的DLL函数，然而这就可能带来各种灾难性的后果，每当偶然情况下用不正确的参数调用该函数时，都可以引发一个GPF，我们可以这样进行严格的类型检查并且帮助防止这类问题。这种方法就是在函数的声明中使用Alias技巧，看看下面的两个声明：`Declare Function LoadCursorByName lib " USER " Alias " LoadCursor " (ByVal hInstance As Integer , ByVal lpCursorName As String) As Integer`和`Declare Function LoadCursorByID Lib " USER " Alias " LoadCursor " (ByVal hInstance As Integer , ByVal lpCursorName As Long) As Integer` `LoadCursorByName`用字符串做`lpCursorName`参数访问DLL函数`LoadCursor`，而`LoadCursorByID`访问同样的DLL函数`LoadCursor`，但是用长整型做`lpCursorName`参数，这两个函数都对`lpCursorName`参数进行严格的类型检查，使Visual Basic能在调用DLL函数之前识别出不正确的变量类型，最大限度地减少引起GPF或者导致系统崩溃的机会。除此之外，使用API函数时运行之前最好先存盘，仔细检查调用API函数的参数与声明的类型是否一致，以及严格检查参数是否有效都能减少引发GPF或者系统崩溃。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