

精通编程系列书



内附光盘

Visual Basic API 编程百例通

魏江江 林少景 编著

科学出版社
Science Press

精通编程系列书

Visual Basic API 编程百例通

魏江江 林少景 编著

科学出版社

2001

内 容 简 介

本书主要是针对希望掌握 Visual Basic API 编程的读者而编写的, 根据不同类别分为 7 章, 分别是窗体、控件、系统、网络、文件、多媒体和设备, 书中给出精心制作的一百个实例, 其中每个例子都有详细的实例演示讲解和重点分析。所有的实例全部用 API 函数实现。本书旨在引导读者全面掌握 Visual Basic API 编程的方法和技巧。

本书附有配套光盘。它提供了全部实例的源程序, 以方便读者查看、调用和修改等。所有的程序都经过精心调试, 以保证运行正确。光盘中还包括了共享软件及源程序, 并且首次提供了“VB API 手册 2001”编程辅助工具, 工具中提供了完整的 API 函数的定义及查询方式。

本书适用于 Visual Basic 语言的初、中级程序设计人员学习使用, 也可供大专院校计算机及相关专业师生阅读。

图书在版编目(CIP)数据

Visual Basic API 编程百例通/魏江江等编著.-北京: 科学出版社, 2001
(精通编程系列书)

ISBN 7-03-009807-2

I.V... II.魏... III.BASIC 语言-程序设计IV.TP312

中国版本图书馆 CIP 数据核字(2001)第 25124 号

1055-62

科学出版社 出版

北京东黄城根北街16号

邮政编码:100717

<http://www.sciencep.com>

双青印刷厂 印刷

科学出版社发行 各地新华书店经销

*

2001 年 10 月第 一 版 开本: 787×1092 1/16

2001 年 10 月第一次印刷 印张: 26 3/4

印数: 1—5 000 字数: 640 000

定价: 39.00 元(含光盘)

(如有印装质量问题,我社负责调换<环伟>)

前 言

目前所有的 Windows 或 NT 编程都必须使用 API，本书主要讲解如何利用 API 函数扩充 Visual Basic 程序功能。本书列举了大量的编程实例，这些实例覆盖了 Visual Basic API 编程的各个方面的知识点，包括了窗体、控件、系统、网络、文件、多媒体和设备等。由于这些实例注重编程的技巧及方法，所以对初学者和中等程度者有实际的应用价值，通过这些分类的实例引导读者能够快速掌握 API 编程的精髓。

本书共分为 7 章。详细内容介绍如下：

第 1 章 窗体

本章主要介绍 API 函数对窗体的操作，其中包括：如何改变窗体标题栏的属性；限制窗体大小及设置窗体在前；控制窗体标题栏的显示、隐藏和闪烁；限制窗体的位置及如何制作“关于”对话框；捕捉各类窗体的功能；窗体热键功能；创建不规则的热区；制作不规则的窗体；实现窗体阴影效果；制作透明的窗体；实现局部透明的窗体；制作并列式的菜单；调用各类系统的文件夹；控制 MDI 窗体。

第 2 章 控件

本章主要介绍 API 函数对控件属性功能的扩展，其中包括：如何控制 ComboBox 右键功能；给 ListBox 的每个列表行加上不同的提示行；控制 ListView 各种属性；改变 ListBox 的行间距；设置文本框为只读属性；操作剪贴板；实现对控件注册；对文本框内的文字滚动浏览；实现文本框只能输入数字的功能；控制下拉框属性；在程序运行过程中移动窗体内的控件或移动无标题栏的窗体；获得文本框的行数或文本框任意行的文字；为 ListBox 设定水平滚动轴。

第 3 章 系统

本章主要介绍 API 函数对系统的控制，其中包括：如何获得屏幕保护设置的信息；更换桌面背景；捕获桌面背景及设置屏幕分辨率；获取桌面图标信息；控制桌面图标的显示和隐藏；操作“开始”菜单中的“文档”列表；查看应用程序的输入密码；改变计算机名称；用程序关闭计算机；最小化所有窗体；判断系统是大字体还是小字体；获取窗体句柄；检测当前机器的货币符号；控制“开始”菜单的弹出速度；设置系统的日期分隔符；设置输入法；设置任务列表；操作磁盘卷标；控制屏幕保护的运行；设置屏幕颜色。

第 4 章 网络

本章主要介绍 API 函数对网络的控制，其中包括：如何获取计算机的 IP 地址；实现电话拨号功能；调用拨号网络；Ping 一个 IP 地址；获得浏览器地址栏中的网址；实现邮件发送功能；实现软件下载功能；操作 IE 的收藏夹；实现驱动器的网络映射；监测和控

制网络；实现对网站浏览的屏蔽。

第 5 章 文件

本章主要介绍 API 函数对文件的处理，其中包括：如何快速地从硬盘中找到文件；建立多级的目录；设置开机后自动运行程序功能；拷贝文件；如何获取目录信息；删除文件功能；获取程序运行的路径；监测由 VB 调用的其他执行文件的运行状况；显示文件的属性窗口；获取临时路径和临时文件；获取短文件名；为文件建立快捷方式；读写 ini 文件；获取文件的建立时间及存取时间；如何修改文件日期。

第 6 章 多媒体

本章主要介绍 API 函数实现多媒体的功能，其中包括：如何制作 CD 播放器；实现播放 MP3 声音格式的文件；实现图片显示的效果；使用程序获取图片的属性；判断两个物件是否相交；实现媒体播放器播放功能；实现动态捕捉画面功能；制作 Midi 键盘；演示与声音相关的处理；一种图片显示的方式；设置桌面图标。

第 7 章 设备

本章主要介绍 API 函数操作设备的功能，其中包括：如何获取 CPU 信息；获取光驱的信息；修改驱动器名称；控制打印机；获取网卡地址；控制特殊键；实现精确记时；控制鼠标；监测鼠标的坐标；设置鼠标的属性；设置键盘的属性；操作遥控杆；显示磁盘剩余空间；对注册表进行操作；机器运行的相关属性；使用未安装的字体；监测用户键盘的操作。

最后，本书与配套光盘是相辅相成的。在配套光盘中提供了详细完整的源程序和编译后的可执行文件，每个源程序都经过 Visual Basic 6.0 编译通过。光盘中需要的相应软件或控件可以在 <http://zmhh.mycool.net> 下载。另外光盘中还为读者提供了一个大型软件的源程序，其中使用了大量的 API 编程技巧；光盘中还特别为读者制作了“VB API 手册 2001”编程辅助工具，其中集成了大量 API 函数的定义及使用方法（包括本书没有涉及到的），以方便读者查询调用，同时还可以通过网络与我们进行编程交流。我们希望能够给读者提供一个立体的学习方式。

由于编者学识有限，难免有不当之处，欢迎广大读者批评指正，欢迎在 <http://zmhh.mycool.net> 论坛与我们进行交流。

作 者

2001 年 5 月于北京

E-Mail:zmhh@263.net

目 录

第 1 章 窗体

第 1 例	标题栏属性.....	2
第 2 例	限制窗体.....	6
第 3 例	标题栏特效.....	11
第 4 例	窗体位置.....	15
第 5 例	捕捉窗体.....	19
第 6 例	设置窗体的热键.....	31
第 7 例	创建不规则热区.....	33
第 8 例	不规则窗体.....	37
第 9 例	窗体阴影效果.....	41
第 10 例	透明窗体.....	45
第 11 例	有洞的窗体.....	50
第 12 例	多列式菜单.....	53
第 13 例	文件夹.....	59
第 14 例	MDI.....	65

第 2 章 控件

第 15 例	控制 ComboBox 右键.....	70
第 16 例	ListBox 提示行.....	74
第 17 例	ListView 特性.....	76
第 18 例	改变 ListBox 的行间距.....	97
第 19 例	设置文本框为只读.....	100
第 20 例	剪贴板.....	102
第 21 例	控件注册.....	104
第 22 例	滚动文本框.....	107
第 23 例	只能输入数字的文本框.....	110
第 24 例	控制下拉框属性.....	112
第 25 例	移动控件.....	115
第 26 例	文本框的信息.....	117
第 27 例	设置 ListBox 水平卷动轴.....	120

第 3 章 系统

第 28 例	屏幕保护的设置.....	124
第 29 例	桌面背景.....	130

第 30 例	屏幕属性.....	133
第 31 例	桌面图标的位置.....	137
第 32 例	控制桌面图标的显隐.....	149
第 33 例	操作文档列表.....	152
第 34 例	查看密码.....	154
第 35 例	改变计算机名称.....	158
第 36 例	关闭计算机.....	161
第 37 例	最小化所有窗体.....	163
第 38 例	系统字体大小.....	165
第 39 例	窗体句柄.....	168
第 40 例	检测当前机器的货币符号.....	170
第 41 例	控制“开始”菜单的弹出速度.....	173
第 42 例	日期分隔符.....	177
第 43 例	输入法.....	182
第 44 例	任务列表.....	187
第 45 例	设置磁盘卷标.....	189
第 46 例	控制屏幕保护.....	193
第 47 例	设置屏幕颜色.....	196

第 4 章 网络

第 48 例	IP 地址.....	202
第 49 例	电话拨号.....	206
第 50 例	拨号网络.....	209
第 51 例	Ping 一个 IP 地址.....	212
第 52 例	浏览器地址栏中的网址.....	221
第 53 例	发送邮件.....	224
第 54 例	下载软件.....	227
第 55 例	加入收藏夹.....	229
第 56 例	网络映射.....	233
第 57 例	网络监测.....	237
第 58 例	网站过滤器.....	242

第 5 章 文件

第 59 例	查找文件.....	250
第 60 例	创建多级目录.....	253
第 61 例	下次开机时自动运行程序.....	256
第 62 例	拷贝文件.....	258
第 63 例	目录信息.....	261
第 64 例	删除文件.....	266

第 65 例	程序运行的路径.....	268
第 66 例	监控程序.....	271
第 67 例	文件属性.....	273
第 68 例	临时路径和临时文件.....	276
第 69 例	短文件名.....	280
第 70 例	建立快捷方式.....	282
第 71 例	读写 ini 文件.....	285
第 72 例	文件的建立时间及存取时间.....	289

第 6 章 多媒体

第 73 例	CD 播放器.....	298
第 74 例	MP3 播放器.....	305
第 75 例	图片转场.....	308
第 76 例	图片属性.....	311
第 77 例	判断两个物件是否相交.....	314
第 78 例	媒体播放器.....	317
第 79 例	透视镜.....	320
第 80 例	Midi 键盘.....	323
第 81 例	声音处理.....	332
第 82 例	图片显示特效.....	337
第 83 例	设置桌面图标.....	340

第 7 章 设备

第 84 例	CPU.....	346
第 85 例	光驱信息.....	350
第 86 例	修改驱动器名称.....	354
第 87 例	控制打印机.....	357
第 88 例	网卡地址.....	361
第 89 例	控制特殊键.....	365
第 90 例	精确记时.....	369
第 91 例	控制鼠标.....	371
第 92 例	鼠标的坐标.....	375
第 93 例	鼠标设置.....	379
第 94 例	键盘设置.....	384
第 95 例	遥控杆.....	388
第 96 例	磁盘剩余空间.....	395
第 97 例	注册表操作.....	398
第 98 例	本机运行状况.....	405
第 99 例	使用未安装的字体.....	409

第 100 例 监测键值	412
附录 A API 函数说明	415
附录 B 配套光盘的使用	419



第

1

章

窗 体

本章主要介绍 API 函数对窗体的操作，其中包括：
改变窗体标题栏的属性；限制窗体大小及设置窗体在前；控制窗体标题栏的显示、隐藏和闪烁；限制窗体的位置及如何制作“关于”对话框；捕捉各类窗体的功能；窗体热键功能；创建不规则的热区；制作不规则的窗体；实现窗体阴影效果；制作透明的窗体；实现局部透明的窗体；制作并列式的菜单；调用各类系统的文件夹；控制 MDI 窗体。

第 1 例 标题栏属性

实例说明

本例演示如何改变窗体标题栏的属性。运行程序，单击“标题栏按钮无效”按钮，标题栏上的按钮将会不起作用；单击“改变标题栏大小”按钮，窗体的标题栏将会变小。

步骤

具体的操作步骤如下：

1. 建一个新工程，将其命名为 example1.vbp，该工程包括如表 1.1 所示的源文件。

表 1.1 源文件

文件名	描 述
Form1.frm	主窗口
Module1.bas	模块，API 函数声明

2. 各窗体对象和属性如表 1.2 所示，窗口界面如图 1.1 所示。

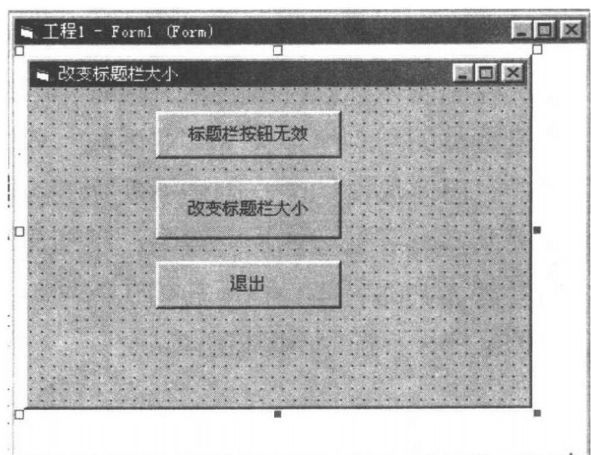


图 1.1 改变标题栏大小的窗口界面

表 1.2 对象和属性表

对 象	属 性	值
Form: 主窗体	名称	Form1
	Caption	改变标题栏大小
	BorderStyle	1
CommandButton	名称	Command1
	Caption	改变标题栏大小

续表

对 象	属 性	值
CommandButton	名称	Command2
	Caption	标题栏按钮无效
CommandButton	名称	Command3
	Caption	退出

3. 程序代码

(1) 在 Module1 模块的声明中加入以下代码:

```
Public Declare Function GetWindowLong Lib "user32" Alias "GetWindowLongA"
(ByVal hwnd As Long, ByVal nIndex As Long) As Long
Public Declare Function SetWindowLong Lib "user32" Alias "SetWindowLongA"
(ByVal hwnd As Long, ByVal nIndex As Long, ByVal dwNewLong As Long) As Long
Public Const WS_EX_TOOLWINDOW = &H80&
Public Const GWL_EXSTYLE = (-20)
Public Declare Function SetWindowPos Lib "user32" (ByVal hwnd As Long, ByVal
hWndInsertAfter As Long, ByVal x As Long, ByVal y As Long, ByVal cx As Long,
ByVal cy As Long, ByVal wFlags As Long) As Long
Public Const SWP_FRAMECHANGED = &H20
Public Const SWP_NOMOVE = &H2
Public Const SWP_NOZORDER = &H4
Public Const SWP_NOSIZE = &H1
Public Declare Function GetSystemMenu Lib "user32" (ByVal hMenu As Long,
ByVal bRevert As Long) As Long
Public Declare Function RemoveMenu Lib "user32" (ByVal hMenu As Long, ByVal
nPosition As Long, ByVal wFlags As Long) As Long
Public Const MF_BYPOSITION = &H400&
Public Const MF_REMOVE = &H1000&
Public Const SC_CLOSE = &HF060
Public Const SC_MAXIMIZE = &HF030
Public Const SC_MINIMIZE = &HF020
```

(2) 在 Form1 窗口的 Command1_Click 处理程序中加入以下代码:

```
Private Sub Command1_Click()
    Dim old_style As Long
    '从指定窗口的结构中取得扩展窗口样式。
    old_style = GetWindowLong(hwnd, GWL_EXSTYLE)
    old_style=SetWindowLong(hwnd,GWL_EXSTYLE,old_styleOr
WS_EX_TOOLWINDOW)
    '为窗口指定一个新位置和状态。
    SetWindowPos hwnd, 0, 0, 0, 100, 100, SWP_FRAMECHANGED Or SWP_NOMOVE
```

Or SWP_NOZORDER Or SWP_NOSIZE

End Sub

(3) 在 Form1 窗口的 Command2_Click 处理程序中加入以下代码:

```
Private Sub Command2_Click()
```

‘使“关闭”按钮无效。

```
RemoveMenu GetSystemMenu(hwnd, 0), SC_CLOSE, MF_REMOVE
```

‘使“窗体最大化按钮”按钮无效。

```
RemoveMenu GetSystemMenu(hwnd, 0), SC_MAXIMIZE, MF_REMOVE
```

‘使“窗体最小化按钮”按钮无效。

```
RemoveMenu GetSystemMenu(hwnd, 0), SC_MINIMIZE, MF_REMOVE
```

End Sub

(4) 在 Form1 窗口的 Command3_Click 处理程序中加入以下代码:

```
Private Sub Command3_Click()
```

End ‘结束程序。

End Sub

技巧

1. 本例使用的 API 函数有 GetWindowLong, SetWindowLong, SetWindowPos, GetSystemMenu 和 RemoveMenu。
2. GetWindowLong 函数可以从指定窗口的结构中取得信息。

声明:

```
Declare Function GetWindowLong Lib "user32" Alias "GetWindowLongA" (ByVal  
hwnd As Long, ByVal nIndex As Long) As Long
```

返回值 Long, 由 nIndex 决定。零表示出错。

参数:

hwnd——Long, 欲为其获取信息的窗口的句柄。

nIndex——Long, 欲取回的信息, 其值如表 1.3 所示。

表 1.3 nIndex 的值

值	说 明
GWL_EXSTYLE	扩展窗口样式
GWL_STYLE	窗口样式
GWL_WNDPROC	该窗口的窗口函数的地址
GWL_HINSTANCE	拥有窗口的实例的句柄
GWL_HWNDPARENT	该窗口之父的句柄
GWL_ID	对话框中一个子窗口的标识符
GWL_USERDATA	由应用程序定义

续表

值	说 明
DWL_DLGPROC	这个窗口的对话框函数地址
DWL_MSGRESULT	在对话框函数中处理的一条消息返回的值
DWL_USER	由应用程序定义

3. SetWindowLong 函数可以在窗口结构中为指定的窗口设置信息。

声明:

```
Declare Function SetWindowLong Lib "user32" Alias "SetWindowLongA" (ByVal hwnd
As Long, ByVal nIndex As Long, ByVal dwNewLong As Long) As Long
```

返回值 Long, 指定数据的前一个值。

参数:

hwnd——Long, 欲为其取得信息的窗口的句柄。

nIndex——Long, 其值如表 1.3 所示。

dwNewLong——Long, 由 nIndex 指定的窗口信息的新值。

4. GetSystemMenu 函数取得指定窗口的系统菜单的句柄。

声明:

```
Declare Function GetSystemMenu Lib "user32" Alias "GetSystemMenu" (ByVal hMenu
As Long, ByVal bRevert As Long) As Long
```

返回值 Long, 如执行成功, 返回系统菜单的句柄; 零表示出错。

参数:

hMenu——Long, 窗口的句柄。

bRevert——Long, 如设为 TRUE, 表示接收原始的系统菜单。

5. RemoveMenu 函数可以删除指定的菜单条目。如果删除的条目属于一个弹出式菜单, 那么这个函数不会同时删除弹出式菜单。首先应该用 GetSubMenu 函数取得弹出式菜单的句柄, 再将其删除。

声明:

```
Declare Function RemoveMenu Lib "user32" Alias "RemoveMenu" (ByVal hMenu As
Long, ByVal nPosition As Long, ByVal wFlags As Long) As Long
```

返回值 Long, 非零表示成功; 零表示失败。

参数:

hMenu——Long, 菜单的句柄。

Nposition —— Long, 欲改变的菜单条目的标识符。如在 wFlags 参数中指定了 MF_BYCOMMAND, 这个参数就代表欲改变的菜单条目的命令 ID。如设置的是 MF_BYPOSITION, 这个参数就代表菜单条目在菜单中的位置(第一个条目的位置为零)。

Wflags——Long, 常数 MF_BYCOMMAND 或 MF_BYPOSITION, 取决于 nPosition 参数。

运行效果

运行程序的结果如图 1.2 和图 1.3 所示。

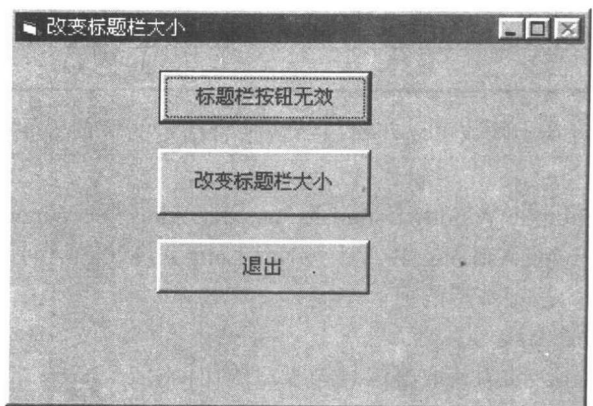


图 1.2 标题栏按钮无效

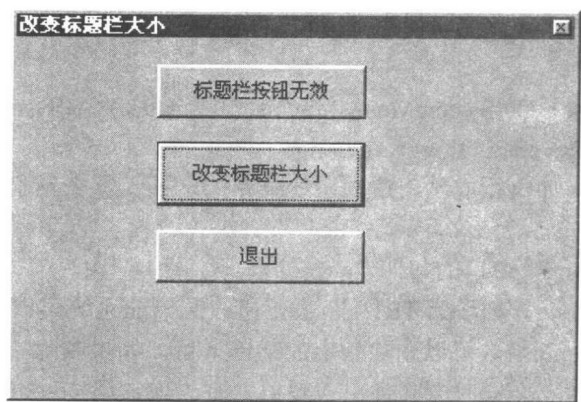


图 1.3 标题栏高度减小

第 2 例 限制窗体

实例说明

本例演示如何限制窗体的大小及设置窗体在前。运行程序，单击“限制窗体最大为 400×400”按钮，这时，窗体只能扩大到一定的范围；单击“窗体在前”按钮，该窗体将永远在其他窗体的前面。

步骤

具体的操作步骤如下：

1. 创建一个新工程，将其命名为 example2.vbp，该工程包括如表 1.4 所示的源文件。

表 1.4 源文件

文件名	描 述
Form1.frm	主窗体
Module1.bas	模块, API 函数声明

2. 各窗体对象和属性如表 1.5 所示, 窗口界面如图 1.4 所示。

表 1.5 对象和属性表

对 象	属 性	值
Form: 主窗体	名称	Form1
	Caption	限制窗体大小
CommandButton	名称	Command1
	Caption	限制窗体最大为 400×400
CommandButton	名称	Command2
	Caption	恢复窗体大小
CommandButton	名称	Command3
	Caption	窗体在前
CommandButton	名称	Command4
	Caption	取消窗体

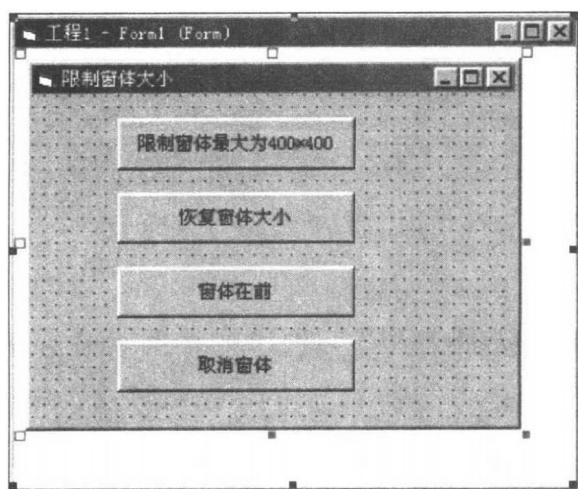


图 1.4 限制窗体的窗体界面

3. 程序代码

(1) 在 **Module1** 模块的声明中加入以下代码:

```
Option Explicit
```

```
Public Declare Function SetWindowLong Lib "user32" Alias "SetWindowLongA"  
(ByVal hwnd As Long, ByVal nIndex As Long, ByVal dwNewLong As Long) As Long  
Public Declare Function GetWindowLong Lib "user32" Alias "GetWindowLongA"  
(ByVal hwnd As Long, ByVal nIndex As Long) As Long
```

```

Public Declare Function CallWindowProc Lib "user32" Alias "CallWindowProcA"
(ByVal lpPrevWndFunc As Long, ByVal hwnd As Long, ByVal Msg As Long, ByVal wParam
As Long, ByVal lParam As Long) As Long

Public Declare Sub CopyMemory Lib "KERNEL32" Alias "RtlMoveMemory" (lpvDest
As Any, lpvSource As Any, ByVal cbCopy As Long)

Public Declare Function SetWindowPos Lib "user32" (ByVal hwnd As Long, ByVal
hWndInsertAfter As Long, ByVal x As Long, ByVal y As Long, ByVal cx As Long,
ByVal cy As Long, ByVal wFlags As Long) As Long

Public Const GWL_hwnd = (-4)
Public Const WM_GETMINMAXINFO = &H24

Type POINTAPI '定义了点的 x 和 y 坐标。
    x As Long
    y As Long
End Type

Type MINMAXINFO '包含了窗口最大尺寸及位置。
    ptReserved As POINTAPI
    ptMaxSize As POINTAPI
    ptMaxPosition As POINTAPI
    ptMinTrackSize As POINTAPI
    ptMaxTrackSize As POINTAPI
End Type

Public Prewinsize As Long

Public Function winsize(ByVal hwnd As Long, ByVal Msg As Long, ByVal wParam
As Long, ByVal lParam As Long) As Long
    Dim lwd As Long, hwd As Long
    Dim maxwin As MINMAXINFO
    If Msg = WM_GETMINMAXINFO Then
        '复制内存区。
        CopyMemory maxwin, ByVal lParam, Len(maxwin)
        maxwin.ptMaxTrackSize.x = 400 '设定最大 Resize 的宽度。
        maxwin.ptMaxTrackSize.y = 400 '设定最大 Resize 的高度。
        CopyMemory ByVal lParam, maxwin, Len(maxwin)
    End If
    '将消息传给窗体。
    winsize = CallWindowProc(Prewinsize, hwnd, Msg, wParam, lParam)
End Function

(2) 在 Form1 窗口的 Command1_Click 处理程序中加入以下代码:
Private Sub Command1_Click()
    Dim ret As Long

```