

80例上手 VB 6 编程

一本不可多得的程序员进阶完全手册

唐凯军 汤惠莉 编著

- 编程高手为你精心打造的80个精彩实例
- 双色印刷使你轻松把握编程技巧
- 赏心悦目的版式设计使你阅读更轻松
- 大量可以重复利用的实例代码
- 一个实例教给你一种新的编程思路
- 学习程序规划设计的最优方案
- API函数的多层次应用与掌握
- 常用控件的学习与应用

超值精美光盘

- 6 全书实例完整源代码——“拿来就用”的现成资源
- 7 全书实例可执行程序——程序执行效果先睹为快
- 6 电脑报·编程乐园精华——众多编程爱好者经验分享
- 7 VB控件——让VB编程更轻松
- VB6自制函数——简单代码大功效
- 编程素材——丰富的图标、音效等素材为你VB 6编程锦上添花
- 7 编程实用问答——为你解答VB 6编程难题



80例上手 VB 6 编程

唐凯军 汤惠莉 编著

江苏工业学院图书馆
藏书章



山东电子音像出版社出版

内容提要

学编程的人都知道，无论哪一种程序语言，都是学会容易学精难！有没有一种好的方法能够使人快速上手呢？本手册就是这样一个桥梁，能够带你快速掌握“程序员修炼之道”！

本手册精心选用生动有趣、涉及面广的VB 6 程序语言实用案例，分为界面设计、系统设置、硬件与控件、文件操作、图形图像多媒体、数据库应用、网络与通讯、趣味游戏8个章节，对编程思路和实例代码进行了深入浅出的详细解析，有助于你通过实例代码掌握编程思想与方法，迅速提高自己的程序设计水平。光盘中还提供了全部实例的源代码，使你可以“拿来就用”！

精彩光盘

- ★全书实例完整源代码——“拿来就用”的现成资源
- ★全书实例可执行程序——程序执行效果先睹为快
- ★电脑报·编程乐园精华——众多编程爱好者经验分享
- ★VB 控件——让VB 编程更轻松
- ★VB6 自制函数——简单代码大功效
- ★编程素材——丰富的图标、音效等素材为你VB 6 编程锦上添花
- ★编程实用问答——为你解答VB 6 编程难题

书 名：80 例上手 VB 6 编程
编 著：唐凯军 汤惠莉
责任编辑：李 萍
执行编辑：邢政义
封面设计：邓玉萍
组版编辑：李品娟
监 制：谢均建
出版单位：山东电子音像出版社
地 址：济南市胜利大街39号
邮政编码：250001
电 话：(0531)2060055-7616
发 行：山东电子音像出版社
经 销：各地新华书店、报刊亭
C D 生产：北京中联鸿远光盘科技发展有限公司
文本印刷：重庆升光电力印务有限公司
开本规格：787mm × 1092mm 1/16 19.75 印张 650 千字
版 本 号：ISBN 7-900398-57-0
版 次：2004 年 10 月第 1 版 2004 年 10 月第 1 次印刷
定 价：28.00 元(1CD+ 手册)

光 盘 导 读

★全书实例完整源代码及可执行程序

全书 80 个实例完整源代码及可执行程序

光盘根目录

★电脑报·编程乐园精华

VB 实例与技巧 (181 篇)
Delphi 实例与技巧 (89 篇)
VC、C++ Builder、Turbo C 实例与技巧 (50 篇)
PowerBuilder 实例与技巧 (25 篇)
VFP 实例与技巧 (31 篇)

编程乐园精华\vb
编程乐园精华\delphi
编程乐园精华\VC
编程乐园精华\PowerBuilder
编程乐园精华\VFP

★VB 控件

acd (播放 CD 的控件, 并能实现 CD 转 AVI)
autodial (自动拨号的控件)
bemidi (播放 MIDI 音乐的控件)
Big2GB (BIG5 码 <=> GB 码互换的控件)
ccrpani (播放 AVI 文件的控件)
ccrpprg (功能强大的进度条控件)
chk1st (将 CheckBox 和 ListBox 合二为一)
dnslookup (查找计算机 IP 地址控件)
dynarec (录音控件)
extrainbutton (RainbowButton 按钮控件)
eyes (跟着鼠标转的眼睛控件)
filetime (一个设置文件时间的控件)
ftpocx (FTP 登录控件)
httpexp (把 MP3 转换为 WAV 文件格式)
keyboard (检测键盘按键控件)
keypress (定义热键的控件)
lcd (模拟 LED 液晶显)
midiamp3 (播放任何声音格式文件的控件)
mp3enc (MP3 播放控件)
Office (Office 风格菜单动态库)
ovalbutton (椭圆型按钮控件)
ProcBar (显示百分比数字进度条控件)
sblist20 (能在列表项中使用颜色、字体、位图)
spelltxt (一个文档拼音检查的控件)
transparent (可使窗体变成透明)
vertmenu (制作类似 Outlook 那样的垂直菜单)
weiqi (围棋对弈程序的开发控件)
winpopup (发送和接收 Winpopup 信息控件)
xzipdemo (Xceed Zip 控件)

VB 控件\acd
VB 控件\autodial.exe
VB 控件\bemidi.exe
VB 控件\big2gb.exe
VB 控件\ccrpani.exe
VB 控件\ccrpprg.exe
VB 控件\chk1st.exe
VB 控件\dnslookup.exe
VB 控件\dynarec.exe
VB 控件\extrainbutton.exe
VB 控件\eyes.exe
VB 控件\filetime.exe
VB 控件\ftpocx.exe
VB 控件\httpexp.exe
VB 控件\keyboard.exe
VB 控件\keypress.exe
VB 控件\lcd.exe
VB 控件\midiamp3.exe
VB 控件\mp3enc.exe
VB 控件\Office.exe
VB 控件\ovalbutton.exe
VB 控件\ProcBar.exe
VB 控件\sblist20.exe
VB 控件\spelltxt.exe
VB 控件\transparent.exe
VB 控件\vertmenu.exe
VB 控件\weiqi.exe
VB 控件\winpopup.exe
VB 控件\xzipdemo.exe

★ VB6 自制函数

删除字符串内某一文字
取代字符串内某一文字
删除字符串内所有空白
为 VB Script 提供 Format 程序
求某月有多少天
得到字符串中指定位置的字符
求最大公因数
在字符串内插入文字
判断字符是否符合文档名格式
判断一对象是否击入另一对象里
超长整数加法
超长整数减法
长乘法
传回经过时间

VB6 自制函数精选 \AllDelStr
VB6 自制函数精选 \AllReStr
VB6 自制函数精选 \ALLTrim
VB6 自制函数精选 \Format
VB6 自制函数精选 \GetDaysofMonthByDate
VB6 自制函数精选 \GetStr
VB6 自制函数精选 \Hcfab
VB6 自制函数精选 \InsStr
VB6 自制函数精选 \IsFile
VB6 自制函数精选 \IsImpact
VB6 自制函数精选 \LnAdd
VB6 自制函数精选 \Mux
VB6 自制函数精选 \Spends
VB6 自制函数精选 \SubAdd

★ 编程素材

图标 ××××

电脑图标
动物图标
人物图标
植物图标
自然图标
爱心图标
箭头图标
交通图标
节日图标
食品图标
手势图标
书写图标
物品图标

素材\图标\电脑
素材\图标\动物
素材\图标\人物
素材\图标\植物
素材\图标\自然
素材\图标\爱心
素材\图标\箭头
素材\图标\交通
素材\图标\节日
素材\图标\食品
素材\图标\手势
素材\图标\书写
素材\图标\物品

音效 ××××

动物效果
交通效果
军事效果
生活效果
娱乐效果
自然效果

素材\音效\动物
素材\音效\交通
素材\音效\军事
素材\音效\生活
素材\音效\娱乐
素材\音效\自然

★ 编程实用问答

7 部分 66 个实用问答

VB6 编程实用问答

目 录

第1章 界面设计

一个好的程序，不仅要有实在的内涵，还应该有美丽的外表！当你编制了一个程序之后，一定希望它能够有漂亮的界面、给它配上美丽的菜单图标、具有与众不同的工具条……这样你的程序才会脱颖而出！

- 1** 实例1 利用 API 函数打造图文菜单
- 3** 实例2 利用 API 函数自定义浮动工具条
- 8** 实例3 利用 SendKeys 函数实现打字效果的文字
- 10** 实例4 利用计时器性能制作自己的屏保
- 13** 实例5 用 API 函数制作异形窗口时钟
- 17** 实例6 利用控制数组制作自己的计算器
- 22** 实例7 打造自我的 MsgBox 信息提示框
- 27** 实例8 利用 API 函数自制多功能 VCD 播放器

第2章 系统设置

很多人都用过《魔法兔子》和《Windows 优化大师》之类的系统修改软件吧！有没有想过自己也设计一个这样的软件来帮自己和朋友解决困难呢？本章通过很多实例教会你这类程序编制的一些简便方法，通过它们，聪明的你一定会触类旁通，作出具有个人特色的系统设置软件！

- 34** 实例1 利用 API 函数修改桌面
- 39** 实例2 利用 API 函数隐藏桌面项目
- 43** 实例3 使用 API 函数制作 URL 快捷方式制造程序
- 46** 实例4 使用 API 函数打造任意修改系统日期的万年历
- 51** 实例5 利用 API 函数制作动态系统分辨率修改程序
- 56** 实例6 利用 API 函数轻松实现输入法切换
- 59** 实例7 利用 API 函数查看、中止木马病毒进程
- 64** 实例8 用 VB 编程来保护你的桌面隐私
- 66** 实例9 用 VB 编写屏幕抓图程序
- 70** 实例10 利用 API 函数获取所需颜色
- 73** 实例11 用 VB 窃取可执行文件内部所有图标
- 77** 实例12 用 VB 打造自己的英语写作伴侣

第3章 硬件与控件

硬件和控件的控制与使用，在编程中应该是一个难点了。很多时候，大家都会在这方面感到不知所措。其实，只要你掌握了其中的奥妙和技巧，就没有什么克服不了的事情。本章的目的就是通过几个小巧的实例，使你顺利掌握这方面的编程技巧。

- 81** 实例1 利用API函数控制CDROM驱动器
- 84** 实例2 利用SHFormatDrive()格式化磁盘
- 86** 实例3 利用Timer制作定时器
- 88** 实例4 利用ActiveX控件管理文件资源
- 93** 实例5 自制完美的LED数字显示控件
- 103** 实例6 用Active控件实现温度计显示拷贝进程
- 105** 实例7 自制温度转换控件
- 108** 实例8 用网格控件自制图标缩略浏览器

第4章 文件操作

可以说，文件构成了计算机软件部分的全部，那么我们就不可避免的要对文件进行修改、比较、编辑……这些要是都是用自己编的软件来实现那该多好！看看本章的内容，相信你一定可以做到！

- 112** 实例1 利用FileSystem Object察看修改文件
- 115** 实例2 运用文本框自制简易记事本
- 119** 实例3 用带格式的文本框自制Word文件处理器
- 122** 实例4 用FileSystemObject自制文件搜捕利器
- 125** 实例5 用访问文件的方法自制文件比较程序
- 128** 实例6 用检索文件关键字的方法自编程检测CIH病毒
- 130** 实例7 利用微软工具做自己的帮助文件
- 133** 实例8 用name命令自制批量文件名修改程序
- 136** 实例9 自制批量图片格式转换器

第5章 图形图像多媒体

用电脑就一定会用到图形图像多媒体软件,可是我们常常只是需要用到其中的一部分内容,要是为了这一小点改动,就使用一些大型软件,是不是感觉有点杀鸡用牛刀呀!其实作为一个程序员,完全可以根据自己的实际需求作出一些简单的软件。

- 140** 实例1 用API函数自制声卡测试程序
- 145** 实例2 用API函数自编程序控制音量大小
- 149** 实例3 利用API函数自制屏幕放大镜
- 153** 实例4 利用API函数自制控制按钮
- 156** 实例5 自制三维立体文字的小工具
- 162** 实例6 利用“取点法”制作自己的个性化图标
- 168** 实例7 用VB模拟PhotoShop滤镜效果
- 174** 实例8 巧用VB使图片二合一
- 178** 实例9 用Image控件自制图片浏览器程序
- 183** 实例10 利用RGB函数自制调色板
- 185** 实例11 利用ShockWare Flash制作多功能Flash播放器
- 189** 实例12 用WMP控件自制多媒体视频播放器

第6章 数据库应用

数据库的使用是和每个人密切相关的,电子通讯录、个人收支统计、学生成绩表等等很多都是使用的数据库来实现的。用了很多网络上的这些软件,你是不是感觉总有地方不合你的意思呀?其实你完全可以根据本章提供的方法,编制一个十全十美的数据库软件。

- 192** 实例1 利用DAO创建数据库结构
- 196** 实例2 利用DAO管理数据库
- 199** 实例3 利用VB进行数据库事务处理
- 202** 实例4 利用ADO进行数据库导出操作
- 205** 实例5 利用DAO进行数据导入
- 208** 实例6 用ADO制作SQL查询器
- 212** 实例7 利用Crystal Reports巧制家庭财务报表
- 216** 实例8 利用DAO制作通讯录
- 220** 实例9 利用MSFlexGrid制作成绩表
- 224** 实例10 妙用SQL自制小小生词本
- 228** 实例11 利用ADO制作家庭书架
- 231** 实例12 利用ADO打造个人数码相册

第7章 网络与通讯

现在没有不上网的程序员吧！要是你上网使用的是你自己编制的浏览器那该多么能够吸引人家的眼球呀！本章不仅教会你编制自己的浏览器，还让你用自己的程序收发邮件、下载文件，甚至让你自己编制远程控制程序，不过你可不要用来做坏事哟！

- 236** 实例1 编写自己的 Internet 浏览器
- 240** 实例2 用 WebBrowser 制作自己的网络搜索软件
- 243** 实例3 利用 VB 提取 HTML 文件中的 E-MAIL 地址
- 246** 实例4 用 WinSock 控件自制邮件发送器
- 249** 实例5 让自己的软件自动在线升级
- 253** 实例6 利用 Inet 控件自制 FlashGet 下载软件
- 256** 实例7 利用 UDP 协议自制点对点聊天 QQ
- 260** 实例8 用 VB 编写自己的代理服务器
- 266** 实例9 用 WinSock 控件自制木马端口探测器
- 268** 实例10 利用 VB 编制远程控制程序（服务端）
- 271** 实例11 利用 VB 编制远程控制程序（客户端）

第8章 趣味游戏

相信大家都喜欢玩游戏吧，有没有想过自己设计一个游戏来进行娱乐呢？本章主要就是通过一系列不同的游戏编制方法，带你进入游戏编程之门。

- 276** 实例1 自制键盘练习游戏
- 281** 实例2 用 VB 编制猜数字游戏
- 284** 实例3 用 VB 控件自制拼图游戏
- 289** 实例4 利用随机数自制福利彩票中奖器
- 290** 实例5 VB 控件打造三连胜游戏
- 293** 实例6 体现游戏最基本技术的球类游戏
- 297** 实例7 自制任天堂小蜜蜂射击游戏
- 304** 实例8 自制手机贪吃蛇游戏

第1章

界面设计

本章要点：

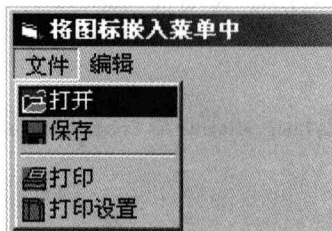
一个好的程序，不仅要有实在的内涵，还应该有美丽的外表！当你编制了一个程序之后，一定希望它能够有漂亮的界面、给它配上美丽的菜单图标、具有与众不同的工具条……这样你的程序才会脱颖而出！本章就是教给你各种界面的编制方法。

实例 1

利用 API 函数打造图文菜单

【程序说明】

在一些流行软件中经常可以见到在菜单中设置位图图标，就像在工具栏中一样。要实现该功能其实并不复杂，下面将为你讲解如何将图标嵌入菜单中。程序效果如图。



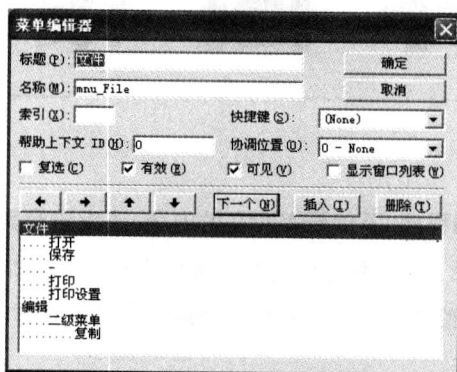
【编程思路】

我们可以使用三个 API 函数来实现该功能。首先通过调用 `GetMenu` 获得主菜单项（文件、编辑）的句柄（Handle），调用 `GetSubMenu` 函数获得下级菜单的句柄，调用 `GetMenu` 函数文件、编辑菜单句柄，`GetSubMenu`（文件、编辑菜单句柄，1）函数返回二级菜单句柄，然后 `GetSubMenu`（文件二级菜单句柄，0）函数返回三级菜单（复制）句柄，最后用 `SetMenuItemBitmaps` 函数将位图设置到相应的菜单项中去。

【编程步骤】

(1) 启动 Visual Basic 6.0, 新建一个标准工程。

(2) 创建菜单。单击菜单“工具/菜单编辑器”, 加入两个主菜单项(文件、编辑), 然后在主菜单项(文件)下设置 4 个二级菜单, 在主菜单项(编辑)下设置一个三级菜单, 菜单项设置如图所示。



(3) 将 Form1 的 Caption 属性设置为“将图标嵌入菜单中”, 在窗体中加入 5 个 Image, 将其属性中的名称分别设置为 imgOpen、imgSave、imgPrint、imgPrintSetup、imgCopy, 在 Picture 上分别插入常见的工具栏图标: 表示打开、保存、打印、打印设置、复制, 并将 Visible 都设置为 False。

【代码解析】

Option Explicit

' API 函数声明

' 用于获取主菜单项句柄

Private Declare Function GetMenu Lib "user32" (ByVal hwnd As Long) As Long

' 用于获取下级菜单句柄

Private Declare Function GetSubMenu Lib "user32" (ByVal hMenu As Long, ByVal nPos As Long) As Long

' 用于将位图设置到菜单项中

Private Declare Function SetMenuItemBitmaps Lib "user32" (ByVal hMenu As Long, ByVal nPosition As Long, ByVal wFlags As Long, ByVal hBitmapUnchecked As Long, ByVal hBitmapChecked As Long) As Long

Const MF_BYPOSITION = &H400&

Private Sub Form_Load()

Dim hMainMenu As Long, hImgSet As Long, sHandle As Long, sHandle2 As Long

' 获取主菜单句柄

hMainMenu = GetMenu(hwnd)

' 获取二级菜单句柄

sHandle = GetSubMenu(hMainMenu, 0)

' 将位图设置到相应的菜单项中

hImgSet = SetMenuItemBitmaps(sHandle, 0, MF_BYPOSITION, imgOpen.Picture, imgOpen.Picture) ' 设置打开菜单

hImgSet = SetMenuItemBitmaps(sHandle, 1, MF_BYPOSITION, imgSave.Picture, imgSave.Picture) ' 设置保存菜单

hImgSet = SetMenuItemBitmaps(sHandle, 3, MF_BYPOSITION, imgPrint.Picture, imgPrint.Picture) ' 设置打印菜单

hImgSet = SetMenuItemBitmaps(sHandle, 4, MF_BYPOSITION, imgPrintSetup.Picture, imgPrintSetup.Picture) ' 设置打印设置菜单

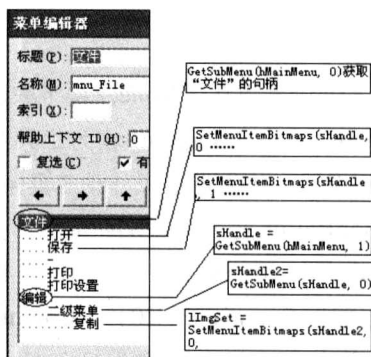
' 获取三级菜单句柄

sHandle = GetSubMenu(hMainMenu, 1) ' 即编辑菜单句柄

```
sHandle2 = GetSubMenu(sHandle, 0) '获取“二级菜单”菜单的下级菜单句柄  
lImgSet = SetMenuItemBitmaps(sHandle2, 0, MF_BYPOSITION, imgCopy.Picture, imgCopy.Picture) '设置复制  
菜单  
End Sub
```

【程序小结】

本例程序中，关键部分是获取菜单项的句柄。下图可以帮助你理解怎样获取菜单句柄及设置位图。

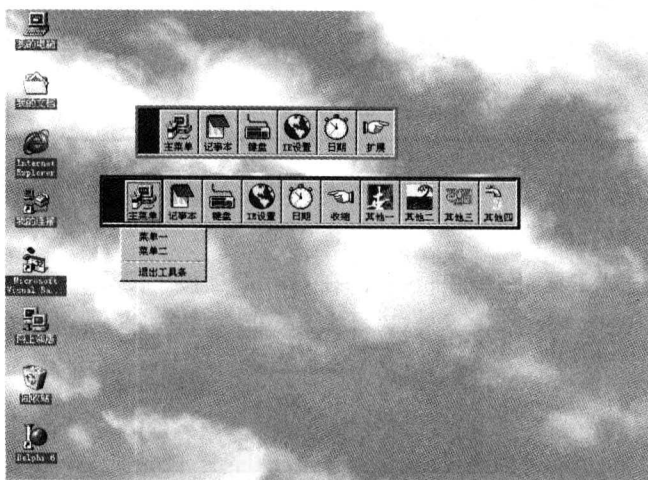


实例 2

利用 API 函数自定义浮动工具条

【程序说明】

用过《东方快译》的读者都会对它的浮动工具菜单印象深刻。这种浮动工具菜单可以放在桌面的任何位置，也可以隐藏在屏幕上方（有点类似 QQ 的界面），使用十分方便。本例就来仿照它做一个浮动工具条，在这个工具条上可以挂接各种常用程序，通过鼠标点击工具条上的按钮来运行，这个特点又比较类似 Windows 的快捷方式。运行效果如图。



【编程思路】

浮动工具条的本质就是一个没有标题的窗口。它的特点在于可以隐藏在屏幕的上、下、左或右边（本例隐藏在屏幕的上边）。这种隐藏就是将窗口的上边界设置成小于 0、下边界小于或等于 0（当窗口隐藏与上边的时候）；窗口的扩展和收缩实际上就是改变窗口的大小。这些功能在 VB 中都能够非常容易地实现。

【编程步骤】

(1) 启动 VB，建立一个标准的 EXE 工程。

(2) 将 Form1 的 BorderStyle 属性设置为 0 - None，接下来添加控件。

在 Form1 上添加一个 PictureBox，名称为 Picture1 作为工具条按钮的框架。

在 Picture1 中添加一个 Label1 控件，作为用鼠标移动工具条时的拖动位置。

在 Picture1 中添加一个 CommonButton 控件，添加完成后，用鼠标右击该控件选复制，然后在 Picture1 中右击鼠标选粘贴，使这个按钮构成一个按钮控件数组，总共粘贴 8 次，这样，这个按钮控件数组共有 9 个元素，名称为 Command1 (0) ~ Command1 (8)。

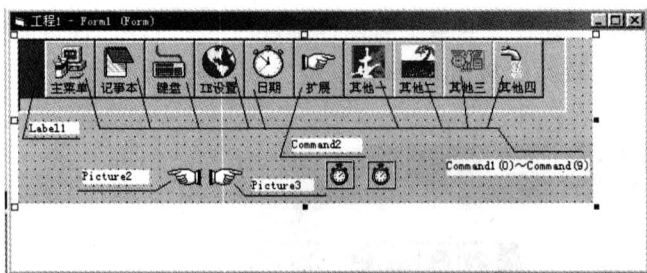
在 Picture1 中另外添加一个 CommandButton，名称为 Command2，将它放在 Command1 控件数组的 Command1 (4) 和 Command1 (5) 之间，这个按钮将作为展开和收缩工具条的功能按钮。

所有按钮控件添加完成后，将它们的 Style 属性改成 1 - Graphical（图形按钮风格）并给除 Command2 以外的所有按钮（修改 Picture 属性）添加上一个合适的图标。

在 Form1 上添加两个 Timer 控件，其中，Timer1 作为 Label1 的闪烁速度控制，Timer2 用来定时检测鼠标是否靠近工具条。

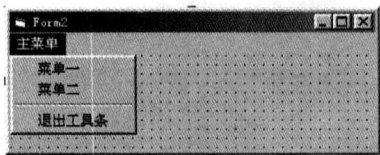
在 Form1 上添加两个 PictureBox 控件，其中 Picture2 用来保存指向左边的手型图标；Picture3 用来保存指向右边的手型图标。

所有控件添加完毕，适当调整一下它们的大小和位置。



(3) 点击“工程 / 添加窗体”，在“新建”标签中选中窗体，按“打开”按钮，系统在工程中自动添加一个 Form2 窗体。

选中 Form2，点击“工具 / 菜单编辑器”，在 Form2 上按下图设计菜单。



其中菜单条的“菜单一”和“菜单二”是演示用的，本例没有什么实际作用，“退出工具条”菜单项目是用来结束程序的。

以上工作做完后，按下表设置各个控件的属性。

控件名称	对象名称	控件属性	设置值
Form	Form1	BorderStyle	0 - None
CommandButton	Command1 (0) ~	Style	1 - Graphical
	Command1 (9)	Caption	根据实际情况设置
	Command2	Picture	Command2 设置成一个指向右边的手型图标，其他按钮根据实际情况设置
PictureBox	Picture2	Picture	指向左边的手型图标
	Picture3	Picture	指向右边的手型图标
	Picture2、Picture3	Visable	False
		BorderStyle	0 - None
Timer	Timer1	Interval	800
	Timer2	Interval	1500

【代码解析】

(1) 声明 API 函数和变量。

```
' 这条 API 用来检测鼠标在屏幕上的位置
Private Declare Function GetCursorPos Lib "user32" (lpPoint As POINTAPI) As Long
' 这条 API 用来将指定窗口放在最前面
Private Declare Function SetWindowPos Lib "user32" (ByVal hwnd As Long, ByVal hWndInsert After As Long, ByVal
x As Long, ByVal Y As Long, ByVal cx As Long, ByVal cy As Long, ByVal wFlags As Long) As Long
' API 函数需要使用的常数
Const HWND_TOPMOST = -1
Const HWND_TOP = 0
Const SWP_SHOWWINDOW = &H40
' 一个结构，用来保存鼠标在屏幕上的位置
Private Type POINTAPI
    x As Long
    Y As Long
End Type
' 一个矩形结构，用来保存工具条左上角和右下角的屏幕坐标
Private Type RECT
    x1 As Integer
    y1 As Integer
    x2 As Integer
    y2 As Integer
End Type
Dim FormRect As RECT ' 定义矩形结构变量
Dim MouseXY As POINTAPI ' 定义保存鼠标在屏幕上位置的变量
Dim Mouse_X As Integer, Mouse_Y As Integer ' 定义鼠标在 Picture 中的位置变量，用来拖动工具条
```

(2) Command1 按钮的事件响应过程，这个过程用来判断用户按下了 Command1 (0) ~ Command1 (9) 中的哪一个按钮，并调用相关程序，实现用工具条模拟 Windows 的快捷方式。本例只定义了 Command1 (0) ~ Command1 (4) 5 个按钮，其他按钮读者可以根据自己的需要定义。

```
Private Sub Command1_Click(Index As Integer)
```

```
' 下面的选择语句用来判断用户按下了哪一个按钮并调用相应程序
```

```
Select Case Index
```

```
Case 0 ' 按下了 Command1 (0) 按钮, 显示 Form2 的菜单
```

```
Form1.PopupMenu Form2.MainMenu, vbPopupMenuLeftAlign, Command1(0).Left, _  
Form1.Height
```

```
Case 1 ' 按下了第二个按钮, 调用记事本程序
```

```
x = Shell("C:\WINDOWS\notepad.exe", vbNormalFocus)
```

```
Case 2 ' 按下了第三个按钮, 调用键盘设置程序
```

```
x = Shell("rundll32.exe shell32.dll,Control_RunDLL access.cpl,,1")
```

```
Case 3 ' 按下了第四个按钮, 调用 IE 的 Option 设置程序
```

```
x = Shell("rundll32.exe shell32.dll,Control_RunDLL inetcpl.cpl")
```

```
Case 4 ' 按下了第五个按钮, 调用日期设置、时间设置程序
```

```
x = Shell("rundll32.exe shell32.dll,Control_RunDLL timedate.cpl")
```

```
Case 5
```

```
' 放入自己的代码
```

```
End Select
```

```
End Sub
```

(3) Command2 事件响应程序, 该程序用来扩展 / 收缩工具条。

```
Private Sub Command2_Click()
```

```
If Command2.Caption = "扩展" Then
```

```
Command2.Caption = "收缩"
```

```
Command2.Picture = Picture2.Picture ' 修改手型图标
```

```
' 将工具条变长, 显示出隐藏的 Command1 (5) ~ Command1 (9) 按钮
```

```
Picture1.Width = 10 * Command1(0).Width + Label1.Width
```

```
Else
```

```
Command2.Caption = "扩展"
```

```
Command2.Picture = Picture3.Picture ' 修改手型图标
```

```
' 将工具条缩短, 隐藏 Command1 (5) ~ Command1 (9) 按钮
```

```
Picture1.Width = 6 * Command1(0).Width + Label1.Width
```

```
End If
```

```
' 同时调整 Form1 的长度, 以适应整个工具条修改后的大小
```

```
Form1.Height = Picture1.Height + 100
```

```
Form1.Width = Picture1.Width + 110
```

```
End Sub
```

(4) 程序启动初始化代码。

```
Private Sub Form_Load()
```

```
' 以下两条语句实现在程序启动的时候将工具条放在屏幕中央
```

```
Form1.Top = (Screen.Height - Form1.Height) / 2
```

```
Form1.Left = (Screen.Width - Form1.Width) / 2
```

```
Dim i As Integer, ControlX
```

```
Picture1.Height = Command1(0).Height + 50
```

```
Picture1.Width = Label1.Width + 6 * Command1(0).Width
```

```
Command2.Width = Command1(0).Width
```

```
Command2.Height = Command1(0).Height
```

```
Label1.Top = 0
```

```
Label1.Left = 0
```

```
Picture1.Left = 50
```

```
Picture1.Top = 50
```

```
ControlX = Label1.Width
```

```
For i = 0 To 8
    Command1(i).Width = 735
    Command1(i).Left = ControlX
    ControlX = ControlX + Command1(0).Width
    If i = 4 Then ControlX = ControlX + Command1(0).Width
Next
Command2.Top = 0
Command2.Left = 5 * Command1(0).Width + Label1.Width
Form1.Height = Picture1.Height + 100
Form1.Width = Picture1.Width + 110
End Sub
```

(5) 窗口重画事件响应代码。

```
Private Sub Form_Paint()
    ' 下面一条语句实现将窗口设置到最前面
    SetWindowPos Form1.hwnd, HWND_TOPMOST, Me.Left \ Screen.TwipsPerPixelX, Me.Top \ Screen.TwipsPerPixelY, _
    Me.Width \ Screen.TwipsPerPixelX, Me.Height \ Screen.TwipsPerPixelY, SWP_SHOW - WINDOW
End Sub
```

(6) 退出工具条。

```
Private Sub Form_Unload(Cancel As Integer)
    End
End Sub
```

(7) Label1 鼠标按下按钮事件响应代码，用来配合 (8) 代码实现移动工具条。

```
Private Sub Label1_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 1 Then ' 判断鼠标是否按下左键
        Mouse_X = X ' 保存鼠标当前坐标
        Mouse_Y = Y
    End If
End Sub
```

(8) Label1 鼠标移动响应代码，用来配合 (7) 实现移动工具条。

```
Private Sub Label1_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    Dim CurrentMouse_X As Single, CurrentMouse_Y As Single
    If Button = 1 Then ' 如果 Button 键赋值为 1 那么执行下一步
        CurrentMouse_X = X - Mouse_X ' 修改鼠标位置 (等于当前位置 - 上一个位置)
        CurrentMouse_Y = Y - Mouse_Y
    ' 修改工具条位置 (等于当前位置 + 鼠标的位移量)
        Form1.Left = Form1.Left + CurrentMouse_X
        Form1.Top = Form1.Top + CurrentMouse_Y
    End If
End Sub
```

(9) Timer1 定时事件响应代码，用来定时改变工具条边框和 Label1 的颜色，实现工具条的闪烁，颜色用随机数产生，范围：0 ~ 16777215，其中：16777215 是 VB 所支持的最大颜色数量。

```
Private Sub Timer1_Timer()
    Form1.BackColor = Rnd * 16777215
    Label1.BackColor = Rnd * 16777215
End Sub
```

(10) Timer2 定时事件响应代码, 用来定时检测鼠标是否靠近了工具条。当工具条被隐藏时, 如果鼠标靠近工具条, 则将工具条从隐藏变成显示。

```
Private Sub Timer2_Timer()  
    GetCursorPos MouseXY '取得鼠标的位置  
    '下面四条语句实现将工具条当前位置的左上角和右下角坐标保存在一个结构中  
    FormRect.x1 = Form1.Left / Screen.TwipsPerPixelX  
    FormRect.y1 = Form1.Top / Screen.TwipsPerPixelY  
    FormRect.x2 = (Form1.Left + Form1.Width) / Screen.TwipsPerPixelX  
    FormRect.y2 = (Form1.Top + Form1.Height) / Screen.TwipsPerPixelY  
    '下一条 IF 语句判断鼠标的位置是否在工具条中  
    If MouseXY.x > FormRect.x1 And MouseXY.x < FormRect.x2 And MouseXY.Y > FormRect.y1 _  
And MouseXY.Y < FormRect.y2 Then  
        '如果工具条已经处于被隐藏状态, 则将工具条显示在屏幕上方  
        If Form1.Top <= 0 Then Form1.Top = Screen.TwipsPerPixelY  
    Else  
        '如果工具条的上边缘进入了屏幕上方, 则将工具条隐藏起来  
        If Form1.Top < 100 Then  
            Form1.Top = -Form1.Height + 40 '隐藏工具条, 但留下一条 40 个屏幕单位的边  
        End If  
    End If  
End Sub
```

【程序小结】

本例没有多少技术难度, 只是使用了几个小技巧:

(1) 移动没有标题栏的窗口: 移动没有标题栏的窗口有多种方法, 比如: 可以用 **SendMessage** API 函数来实现。本例是采用的直接捕获鼠标的 **Move** 和 **Down** 事件, 然后修改窗口的 **Top** 和 **Left** 来实现的。

(2) 窗口边框的闪烁效果: 在窗口上放置一个图片框, 这个图片框并不完全填满整个窗口, 而是与窗口保留较小的边缘, 这个边缘看起来就象一个矩形边框, 只要定时修改窗口的背景颜色, 就实现了窗口“边框”的闪烁。

(3) 窗口的隐藏: 修改窗口的 **Top** 为小于 0, 用 **Form1.Top = -Form1.Height + 40** 语句实现。

实例 3

利用 SendKeys 函数实现打字效果的文字

【程序说明】

本例使用 **SendKeys** 命令实现字符显示的打字效果, 具体运行效果如图。