

Windows API Reference for Visual Basic

This file contains definitions of functions, constants and data structures for use within Visual Basic. It is based on the Microsoft WINAPI.TXT file with additions and corrections included in Microsoft Knowledge Base article Q74526, dated March 9, 1992. This file copyright (c)1992 Stephen Pruitt. Send corrections, comments or suggestions to the author through CompuServe PPN 70244,365.

Visual Basic, Windows and WINAPI.TXT are copyright (c)1991 Microsoft Corporation. The WINAPI.TXT file is included with permission of Microsoft Corp. Microsoft has no warranty or obligation and is not responsible for any errors in this document.

WARNING: This file should **only** be used in conjunction with the Microsoft Windows Programmer's Reference manual. Some functions can not be used within Visual Basic, and others can cause your system to crash or behave strangely!

This file includes data definitions from WINAPI.TXT which could not be related to any functions (see General Purpose Data Definitions, below) and constants with unknown values (see Printer Control Functions). Information regarding these items would be appreciated.

INSTRUCTIONS FOR USING THIS HELP FILE:

You can locate the information you want through two ways:

SEARCH for a desired function, constant or structure by name

Page or scroll down this screen to the menu and select the desired category of functions

The menu will bring you to the category of functions. If there are a small number they are listed on that screen otherwise there are selections for logical groups of definitions. Search brings you directly to the requested definitions. All definitions are grouped in accordance with the Microsoft Windows Programmer's Reference.

To copy the desired definitions into your Visual Basic program, use the EDIT-COPY menu selection (or press CTL-INS) while the desired data is displayed, then switch back to the Visual Basic programming box and use EDIT-INSERT (or SHIFT-INS) to insert the text.

IMPORTANT: Delete any unneeded definitions, as they will increase the size of your program.

The easiest way to use this file is to reduce the size so both it and the Visual Basic box can be seen at once, then copy all of the needed definitions in succession. This file can be kept minimized at other times to provide faster access.

NOTE: You can load this file from Visual Basic by calling for HELP from VB then using the FILE-OPEN menu commands within Help.

MENU OF FUNCTION CATEGORIES

General Purpose Data Definitions

[Constants](#)

[Structures](#)

[BitMap Structures](#)

[Unknown/Unreferenced Data](#)

These are data definitions in VBAPI.TXT that are not described in the Reference manual or referenced elsewhere.

Window Manager

- [Message](#)
- [Window Creation](#)
- [Display and Movement](#)
- [Input](#)
- [Hardware](#)
- [Painting](#)
- [Dialog Box](#)
- [Scrolling](#)
- [Menu](#)
- [Information](#)
- [System](#)
- [Clipboard](#)
- [Error](#)
- [Caret](#)
- [Cursor](#)
- [Hook](#)
- [Property](#)
- [Rectangle](#)

Graphics Device Interface

- [Device Context](#)
- [Drawing Tool](#)
- [Color Palette](#)
- [Drawing Attribute](#)
- [Mapping](#)
- [Coordinate](#)
- [Region](#)
- [Clipping](#)
- [Line Output](#)
- [Ellipse and Polygon](#)
- [Bitmap](#)
- [Text](#)
- [Font](#)
- [Metafile](#)
- [Printer Control](#)
- [Printer Escape](#)
- [Environment](#)

System Services

- [Module Management](#)
- [Memory Management](#)
- [Segment](#)
- [Operating System Interrupts](#)
- [Task](#)
- [Resource Management](#)
- [String Manipulation](#)
- [Atom Management](#)
- [Initialization File](#)
- [Communications](#)
- [Sound](#)
- [Utility Functions and Macros](#)
- [File I/O](#)
- [Debugging](#)
- [Optimization Tool](#)

Application Execution

MultiMedia Extensions

..This section not written yet

'General Purpose Constants

Global Const NULL = 0&

'General Purpose Structures

```
Type FARPROC          'used by Callback routines; not supported by VB
    lpFunc As Double   'far pointer to function
End Type
```

```
Type RECT
    left As Integer
    top As Integer
    right As Integer
    bottom As Integer
End Type
```

```
Type POINTAPI
    x As Integer
    y As Integer
End Type
```

```
' Message structure
Type MSG
    hwnd As Integer
    message As Integer
    wParam As Integer
    lParam As Long
    time As Long
    pt As POINTAPI
End Type
```

```
Type HANDLETABLE
    objectHandle As String * 512 ' Array length is arbitrary; may be changed
End Type
```

'BitMap Structures

' Bitmap Header Definition

Type BITMAP '14 bytes

 bmType As Integer
 bmWidth As Integer
 bmHeight As Integer
 bmWidthBytes As Integer
 bmPlanes As String * 1
 bmBitsPixel As String * 1
 bmBits As Long

End Type

Type RGBTRIPLE

 rgbtBlue As String * 1
 rgbtGreen As String * 1
 rgbtRed As String * 1

End Type

Type RGBQUAD

 rgbBlue as String * 1
 rgbGreen As String * 1
 rgbRed As String * 1
 rgbReserved As String * 1

End Type

' structures for defining DIBs

Type BITMAPCOREHEADER '12 bytes

 bcSize as Long
 bcWidth As Integer
 bcHeight As Integer
 bcPlanes As Integer
 bcBitCount As Integer

End Type

Type BITMAPINFOHEADER '40 bytes

 biSize As Long
 biWidth As Long
 biHeight As Long
 biPlanes As Integer
 biBitCount As Integer
 biCompression As Long
 biSizeImage As Long
 biXPelsPerMeter As Long
 biYPelsPerMeter As Long
 biClrUsed As Long
 biClrImportant As Long

End Type

' constants for the biCompression field

Global Const BI_RGB = 0&

Global Const BI_RLE8 = 1&

Global Const BI_RLE4 = 2&

Type BITMAPINFO

 bmiHeader as BITMAPINFOHEADER

bmiColors As String * 128 ' Array length is arbitrary; may be changed
End Type

Type BITMAPCOREINFO

 bmciHeader As BITMAPCOREHEADER

 bmciColors As String * 96 ' Array length is arbitrary; may be changed

End Type

Type BITMAPFILEHEADER

 bfType As Integer

 bfSize As Long

 bfReserved1 As Integer

 bfReserved2 As Integer

 bfOffBits As Long

End Type

'Undefined Data

' These definitions are found in the WINAPI.TXT file, but the related functions can not be identified.

Global Const ORD_LANGDRIVER = 1 ' The ordinal number for the entry point of
' language drivers.

Global Const ASPECT_FILTERING = &H1

Message Functions

Function Declarations

System Window Messages

User Window Messages

Other Data

MSG Structure is located in General Purpose Structures

'Windows Message Function Definitions

```
Declare Function GetMessage Lib "User" (lpMsg As MSG, ByVal hWnd As Integer, ByVal wParamFilterMin As Integer, ByVal wParamFilterMax As Integer) As Integer
Declare Function TranslateMessage Lib "User" (lpMsg As MSG) As Integer
Declare Function DispatchMessage Lib "User" (lpMsg As MSG) As Long
Declare Function PeekMessage Lib "User" (lpMsg As MSG, ByVal hWnd As Integer, ByVal wParamFilterMin As Integer, ByVal wParamFilterMax As Integer, ByVal wRemoveMsg As Integer) As Integer
Declare Sub PostQuitMessage Lib "User" (ByVal nExitCode As Integer)
Declare Function GetMessagePos Lib "User" () As Long
Declare Function GetMessageTime Lib "User" () As Long
Declare Function SendMessage Lib "User" (ByVal hWnd As Integer, ByVal wParam As Integer, ByVal lParam As Integer, lParam As Any) As Long
Declare Function PostMessage Lib "User" (ByVal hWnd As Integer, ByVal wParam As Integer, ByVal lParam As Integer, lParam As Any) As Integer
Declare Function PostAppMessage Lib "User" (ByVal hTask As Integer, ByVal wParam As Integer, ByVal lParam As Integer, lParam As Any) As Integer
Declare Sub ReplyMessage Lib "User" (ByVal lReply As Long)
Declare Sub WaitMessage Lib "User" ()
Declare Function InSendMessage Lib "User" () As Integer
Declare Function TranslateAccelerator Lib "User" (ByVal hWnd As Integer, ByVal hAccTable As Integer, lpMsg As MSG) As Integer
Declare Function TranslateMDISysAccel Lib "User" (ByVal hWndClient As Integer, lpMsg As MSG) As Integer
Declare Function SetMessageQueue Lib "User" (ByVal cMsg As Integer) As Integer
Declare Function RegisterWindowMessage Lib "User" (ByVal lpString As String) As Integer
```

'Window Messages

Global Const WM_NULL = &H0
Global Const WM_CREATE = &H1
Global Const WM_DESTROY = &H2
Global Const WM_MOVE = &H3
Global Const WM_SIZE = &H5
Global Const WM_ACTIVATE = &H6
Global Const WM_SETFOCUS = &H7
Global Const WM_KILLFOCUS = &H8
Global Const WM_ENABLE = &HA
Global Const WM_SETREDRAW = &HB
Global Const WM_SETTEXT = &HC
Global Const WM_GETTEXT = &HD
Global Const WM_GETTEXTLENGTH = &HE
Global Const WM_PAINT = &HF
Global Const WM_CLOSE = &H10
Global Const WM_QUERYENDSESSION = &H11
Global Const WM_QUIT = &H12
Global Const WM_QUERYOPEN = &H13
Global Const WM_ERASEBKGD = &H14
Global Const WM_SYSCOLORCHANGE = &H15
Global Const WM_ENDSESSION = &H16
Global Const WM_SHOWWINDOW = &H18
Global Const WM_CTLCOLOR = &H19
Global Const WM_WININICHANGE = &H1A
Global Const WM_DEVMODECHANGE = &H1B
Global Const WM_ACTIVATEAPP = &H1C
Global Const WM_FONTCHANGE = &H1D
Global Const WM_TIMECHANGE = &H1E
Global Const WM_CANCELMODE = &H1F
Global Const WM_SETCURSOR = &H20
Global Const WM_MOUSEACTIVATE = &H21
Global Const WM_CHILDACTIVATE = &H22
Global Const WM_QUEUESYNC = &H23
Global Const WM_GETMINMAXINFO = &H24
Global Const WM_PAINTICON = &H26
Global Const WM_ICONERASEBKGD = &H27
Global Const WM_NEXTDLGCTL = &H28
Global Const WM_SPOOLERSTATUS = &H2A
Global Const WM_DRAWITEM = &H2B
Global Const WM_MEASUREITEM = &H2C
Global Const WM_DELETEITEM = &H2D
Global Const WM_VKEYTOITEM = &H2E
Global Const WM_CHARTOITEM = &H2F
Global Const WM_SETFONT = &H30
Global Const WM_GETFONT = &H31
Global Const WM_QUERYDRAGICON = &H37
Global Const WM_COMPAREITEM = &H39
Global Const WM_COMPACTING = &H41
Global Const WM_NCCREATE = &H81
Global Const WM_NCDESTROY = &H82
Global Const WM_NCCALCSIZE = &H83
Global Const WM_NCHITTEST = &H84
Global Const WM_NCPAINT = &H85
Global Const WM_NCACTIVATE = &H86

Global Const WM_GETDLGCODE = &H87
Global Const WM_NCMOUSEMOVE = &HA0
Global Const WM_NCLBUTTONDOWN = &HA1
Global Const WM_NCLBUTTONUP = &HA2
Global Const WM_NCLBUTTONDBLCLK = &HA3
Global Const WM_NCRBUTTONDOWN = &HA4
Global Const WM_NCRBUTTONUP = &HA5
Global Const WM_NCRBUTTONDBLCLK = &HA6
Global Const WM_NCMBUTTONDOWN = &HA7
Global Const WM_NCMBUTTONUP = &HA8
Global Const WM_NCMBUTTONDBLCLK = &HA9

Global Const WM_KEYFIRST = &H100
Global Const WM_KEYDOWN = &H100
Global Const WM_KEYUP = &H101
Global Const WM_CHAR = &H102
Global Const WM_DEADCHAR = &H103
Global Const WM_SYSKEYDOWN = &H104
Global Const WM_SYSKEYUP = &H105
Global Const WM_SYSCHAR = &H106
Global Const WM_SYSDEADCHAR = &H107
Global Const WM_KEYLAST = &H108

Global Const WM_INITDIALOG = &H110
Global Const WM_COMMAND = &H111
Global Const WM_SYSCOMMAND = &H112
Global Const WM_TIMER = &H113
Global Const WM_HSCROLL = &H114
Global Const WM_VSCROLL = &H115
Global Const WM_INITMENU = &H116
Global Const WM_INITMENUPOPUP = &H117
Global Const WM_MENUSELECT = &H11F
Global Const WM_MENUCHAR = &H120
Global Const WM_ENTERIDLE = &H121

Global Const WM_MOUSEFIRST = &H200
Global Const WM_MOUSEMOVE = &H200
Global Const WM_LBUTTONDOWN = &H201
Global Const WM_LBUTTONUP = &H202
Global Const WM_LBUTTONDBLCLK = &H203
Global Const WM_RBUTTONDOWN = &H204
Global Const WM_RBUTTONUP = &H205
Global Const WM_RBUTTONDBLCLK = &H206
Global Const WM_MBUTTONDOWN = &H207
Global Const WM_MBUTTONUP = &H208
Global Const WM_MBUTTONDBLCLK = &H209
Global Const WM_MOUSELAST = &H209

Global Const WM_PARENTNOTIFY = &H210
Global Const WM_MDICREATE = &H220
Global Const WM_MDIDESTROY = &H221
Global Const WM_MDIACTIVATE = &H222
Global Const WM_MDIRESTORE = &H223
Global Const WM_MDIINEXT = &H224
Global Const WM_MDIMAXIMIZE = &H225
Global Const WM_MDITILE = &H226

Global Const WM_MDICASCADE = &H227
Global Const WM_MDIICONARRANGE = &H228
Global Const WM_MDIGETACTIVE = &H229
Global Const WM_MDISETMENU = &H230

Global Const WM_CUT = &H300
Global Const WM_COPY = &H301
Global Const WM_PASTE = &H302
Global Const WM_CLEAR = &H303
Global Const WM_UNDO = &H304
Global Const WM_RENDERFORMAT = &H305
Global Const WM_RENDERALLFORMATS = &H306
Global Const WM_DESTROYCLIPBOARD = &H307
Global Const WM_DRAWCLIPBOARD = &H308
Global Const WM_PAINTCLIPBOARD = &H309
Global Const WM_VSCROLLCLIPBOARD = &H30A
Global Const WM_SIZECLIPBOARD = &H30B
Global Const WM_ASKCBFORMATNAME = &H30C
Global Const WM_CHANGECHAIN = &H30D
Global Const WM_HSCROLLCLIPBOARD = &H30E
Global Const WM_QUERYNEWPALETTE = &H30F
Global Const WM_PALETTEISCHANGING = &H310
Global Const WM_PALETTECHANGED = &H311

' NOTE: All Message Numbers below 0x0400 are RESERVED.

' WM_MOUSEACTIVATE Return Codes

Global Const MA_ACTIVATE = 1
Global Const MA_ACTIVATEANDEAT = 2
Global Const MA_NOACTIVATE = 3

' User Button Notification Codes

Global Const BN_CLICKED = 0
Global Const BN_PAINT = 1
Global Const BN_HILITE = 2
Global Const BN_UNHILITE = 3
Global Const BN_DISABLE = 4
Global Const BN_DOUBLECLICKED = 5

' Scroll Bar Commands

Global Const SB_LINEUP = 0
Global Const SB_LINEDOWN = 1
Global Const SB_PAGEUP = 2
Global Const SB_PAGEDOWN = 3
Global Const SB_THUMBPOSITION = 4
Global Const SB_THUMBTRACK = 5
Global Const SB_TOP = 6
Global Const SB_BOTTOM = 7
Global Const SB_ENDSCROLL = 8

' Owner draw control types

Global Const ODT_MENU = 1
Global Const ODT_LISTBOX = 2
Global Const ODT_COMBOBOX = 3
Global Const ODT_BUTTON = 4

' Owner draw actions

Global Const ODA_DRAWENTIRE = &H1
Global Const ODA_SELECT = &H2
Global Const ODA_FOCUS = &H4

' Owner draw state

Global Const ODS_SELECTED = &H1
Global Const ODS_GRAYED = &H2
Global Const ODS_DISABLED = &H4
Global Const ODS_CHECKED = &H8
Global Const ODS_FOCUS = &H10

' WinWhere() Area Codes

Global Const HTERROR = (-2)
Global Const HTTRANSPARENT = (-1)
Global Const HTNOWHERE = 0
Global Const HTCLIENT = 1
Global Const HTCAPTION = 2
Global Const HTSYSTEMMENU = 3
Global Const HTGROWBOX = 4
Global Const HTSIZE = HTGROWBOX
Global Const HTMENU = 5
Global Const HTHSCROLL = 6
Global Const HTVSCROLL = 7
Global Const HTREDUCE = 8
Global Const HTZOOM = 9
Global Const HTLEFT = 10
Global Const HTRIGHT = 11
Global Const HTTOP = 12
Global Const HTTOPLEFT = 13
Global Const HTTOPRIGHT = 14
Global Const HTBOTTOM = 15
Global Const HTBOTTOMLEFT = 16
Global Const HTBOTTOMRIGHT = 17
Global Const HTSIZEFIRST = HTLEFT
Global Const HTSIZELAST = HTBOTTOMRIGHT

' Size Message Commands

Global Const SIZENORMAL = 0
Global Const SIZEICONIC = 1
Global Const SIZEFULLSCREEN = 2
Global Const SIZEZOOMSHOW = 3
Global Const SIZEZOOMHIDE = 4

' Key State Masks for Mouse Messages

Global Const MK_LBUTTON = &H1
Global Const MK_RBUTTON = &H2
Global Const MK_SHIFT = &H4
Global Const MK_CONTROL = &H8
Global Const MK_MBUTTON = &H10

' Identifiers for the WM_SHOWWINDOW message

Global Const SW_PARENTCLOSING = 1
Global Const SW_OTHERZOOM = 2
Global Const SW_PARENTOPENING = 3
Global Const SW_OTHERUNZOOM = 4

' WM_SYNC_TASK Commands
Global Const ST_BEGINSWP = 0
Global Const ST_ENDSWP = 1

'User Window Messages

' Private Window Messages Start Here:

Global Const WM_USER = &H400

' Listbox messages

Global Const LB_ADDSTRING = (WM_USER+1)

Global Const LB_INSERTSTRING = (WM_USER+2)

Global Const LB_DELETESTRING = (WM_USER+3)

Global Const LB_RESETCONTENT = (WM_USER+5)

Global Const LB_SETSEL = (WM_USER+6)

Global Const LB_SETCURSEL = (WM_USER+7)

Global Const LB_GETSEL = (WM_USER+8)

Global Const LB_GETCURSEL = (WM_USER+9)

Global Const LB_GETTEXT = (WM_USER+10)

Global Const LB_GETTEXTLEN = (WM_USER+11)

Global Const LB_GETCOUNT = (WM_USER+12)

Global Const LB_SELECTSTRING = (WM_USER+13)

Global Const LB_DIR = (WM_USER+14)

Global Const LB_GETTOPINDEX = (WM_USER+15)

Global Const LB_FINDSTRING = (WM_USER+16)

Global Const LB_GETSELCOUNT = (WM_USER+17)

Global Const LB_GETSELITEMS = (WM_USER+18)

Global Const LB_SETTABSTOPS = (WM_USER+19)

Global Const LB_GETHORIZONTALEXTENT = (WM_USER+20)

Global Const LB_SETHORIZONTALEXTENT = (WM_USER+21)

Global Const LB_SETCOLUMNWIDTH = (WM_USER+22)

Global Const LB_SETTOPINDEX = (WM_USER+24)

Global Const LB_GETITEMRECT = (WM_USER+25)

Global Const LB_GETITEMDATA = (WM_USER+26)

Global Const LB_SETITEMDATA = (WM_USER+27)

Global Const LB_SELITEMRANGE = (WM_USER+28)

Global Const LB_MSGMAX = (WM_USER+33)

Global Const DM_GETDEFID = WM_USER+0

Global Const DM_SETDEFID = WM_USER+1

' Combo Box messages

Global Const CB_GETEDITSEL = (WM_USER+0)

Global Const CB_LIMITTEXT = (WM_USER+1)

Global Const CB_SETEDITSEL = (WM_USER+2)

Global Const CB_ADDSTRING = (WM_USER+3)

Global Const CB_DELETESTRING = (WM_USER+4)

Global Const CB_DIR = (WM_USER+5)

Global Const CB_GETCOUNT = (WM_USER+6)

Global Const CB_GETCURSEL = (WM_USER+7)

Global Const CB_GETLBTEXT = (WM_USER+8)

Global Const CB_GETLBTEXTLEN = (WM_USER+9)

Global Const CB_INSERTSTRING = (WM_USER+10)

Global Const CB_RESETCONTENT = (WM_USER+11)

Global Const CB_FINDSTRING = (WM_USER+12)

Global Const CB_SELECTSTRING = (WM_USER+13)

Global Const CB_SETCURSEL = (WM_USER+14)

Global Const CB_SHOWDROPDOWN = (WM_USER+15)

Global Const CB_GETITEMDATA = (WM_USER+16)

Global Const CB_SETITEMDATA = (WM_USER+17)

Global Const CB_GETDROPPEDCONTROLRECT = (WM_USER+18)

Global Const CB_MSGMAX = (WM_USER+19)

Global Const DM_GETDEFID = WM_USER+0

Global Const DM_SETDEFID = WM_USER+1

' Edit Control Messages

Global Const EM_GETSEL = WM_USER+0

Global Const EM_SETSEL = WM_USER+1

Global Const EM_GETRECT = WM_USER+2

Global Const EM_SETRECT = WM_USER+3

Global Const EM_SETRECTNP = WM_USER+4

Global Const EM_SCROLL = WM_USER+5

Global Const EM_LINESCROLL = WM_USER+6

' WM_USER+7 is not defined

Global Const EM_GETMODIFY = WM_USER+8

Global Const EM_SETMODIFY = WM_USER+9

Global Const EM_GETLINECOUNT = WM_USER+10

Global Const EM_LINEINDEX = WM_USER+11

Global Const EM_SETHANDLE = WM_USER+12

Global Const EM_GETHANDLE = WM_USER+13

Global Const EM_GETTHUMB = WM_USER+14

Global Const EM_LINELENGTH = WM_USER+17

Global Const EM_REPLACESEL = WM_USER+18

Global Const EM_SETFONT = WM_USER+19

Global Const EM_GETLINE = WM_USER+20

Global Const EM_LIMITTEXT = WM_USER+21

Global Const EM_CANUNDO = WM_USER+22

Global Const EM_UNDO = WM_USER+23

Global Const EM_FMTLINES = WM_USER+24

Global Const EM_LINEFROMCHAR = WM_USER+25

Global Const EM_SETWORDBREAK = WM_USER+26

Global Const EM_SETTABSTOPS = WM_USER+27

Global Const EM_SETPASSWORDCHAR = WM_USER+28

Global Const EM_EMPTYUNDOBUFFER = WM_USER+29

Global Const EM_MSGMAX = WM_USER+30

' Button Control Messages

Global Const BM_GETCHECK = WM_USER+0

Global Const BM_SETCHECK = WM_USER+1

Global Const BM_GETSTATE = WM_USER+2

Global Const BM_SETSTATE = WM_USER+3

Global Const BM_SETSTYLE = WM_USER+4

' Edit Control Notification Codes

Global Const EN_SETFOCUS = &H100

Global Const EN_KILLFOCUS = &H200

Global Const EN_CHANGE = &H300

Global Const EN_UPDATE = &H400

Global Const EN_ERRSPACE = &H500

Global Const EN_MAXTEXT = &H501

Global Const EN_HSCROLL = &H601

Global Const EN_VSCROLL = &H602

' System Menu Command Values

Global Const SC_SIZE = &HF000

Global Const SC_MOVE = &HF010

Global Const SC_MINIMIZE = &HF020
Global Const SC_MAXIMIZE = &HF030
Global Const SC_NEXTWINDOW = &HF040
Global Const SC_PREVWINDOW = &HF050
Global Const SC_CLOSE = &HF060
Global Const SC_VSCROLL = &HF070
Global Const SC_HSCROLL = &HF080
Global Const SC_MOUSEMENU = &HF090
Global Const SC_KEYMENU = &HF100
Global Const SC_ARRANGE = &HF110
Global Const SC_RESTORE = &HF120
Global Const SC_TASKLIST = &HF130
Global Const SC_ICON = SC_MINIMIZE
Global Const SC_ZOOM = SC_MAXIMIZE

'Other Windows Message Data Definitions

' PeekMessage() Options

Global Const PM_NOREMOVE = &H0

Global Const PM_REMOVE = &H1

Global Const PM_NOYIELD = &H2

' MEASUREITEMSTRUCT for ownerdraw

Type MEASUREITEMSTRUCT

 CtlType As Integer

 CtlID As Integer

 itemID As Integer

 itemWidth As Integer

 itemHeight As Integer

 itemData As Long

End Type

' DRAWITEMSTRUCT for ownerdraw

Type DRAWITEMSTRUCT

 CtlType As Integer

 CtlID As Integer

 itemID As Integer

 itemAction As Integer

 itemState As Integer

 hwndItem As Integer

 hDC As Integer

 rcItem As RECT

 itemData As Long

End Type

' DELETEITEMSTRUCT for ownerdraw

Type DELETEITEMSTRUCT

 CtlType As Integer

 CtlID As Integer

 itemID As Integer

 hwndItem As Integer

 itemData As Long

End Type

' COMPAREITEMSTRUCT for ownerdraw sorting

Type COMPAREITEMSTRUCT

 CtlType As Integer

 CtlID As Integer

 hwndItem As Integer

 itemID1 As Integer

 itemData1 As Long

 itemID2 As Integer

 itemData2 As Long

End Type

' Message Structure used in Journaling

Type EVENTMSG

 message As Integer

 paramL As Integer

 paramH As Integer

 time As Long

End Type

Window Creation

Create/Modify Window Functions

Window Class Functions

Define Windows Processing Functions

Structures and Constants

'Create/Modify Windows

Declare Function CreateWindow% Lib "User" (ByVal lpClassName\$, ByVal lpWindowName\$, ByVal dwStyle&, ByVal X%, ByVal Y%, ByVal nWidth%, ByVal nHeight%, ByVal hWndParent%, ByVal hMenu%, ByVal hInstance%, ByVal lpParam\$)

Declare Function CreateWindowEx% Lib "User" (ByVal dwExStyle&, ByVal lpClassName\$, ByVal lpWindowName\$, ByVal dwStyle&, ByVal X%, ByVal Y%, ByVal nWidth%, ByVal nHeight%, ByVal hWndParent%, ByVal hMenu%, ByVal hInstance%, ByVal lpParam\$)

Declare Sub AdjustWindowRect Lib "User" (lpRect As RECT, ByVal dwStyle As Long, ByVal bMenu As Integer)

Declare Sub AdjustWindowRectEx Lib "User" (lpRect As RECT, ByVal dsStyle As Long, ByVal bMenu As Integer, ByVal dwEsStyle As Long)

Declare Function DestroyWindow Lib "User" (ByVal hWnd As Integer) As Integer

Declare Function GetWindowWord Lib "User" (ByVal hWnd As Integer, ByVal nIndex As Integer) As Integer

Declare Function SetWindowWord Lib "User" (ByVal hWnd As Integer, ByVal nIndex As Integer, ByVal wNewWord As Integer) As Integer

Declare Function GetWindowLong Lib "User" (ByVal hWnd As Integer, ByVal nIndex As Integer) As Long

Declare Function SetWindowLong Lib "User" (ByVal hWnd As Integer, ByVal nIndex As Integer, ByVal dwNewLong As Long) As Long

Declare Function GetLastActivePopup Lib "User" (ByVal hWndOwner As Integer) As Integer

Declare Function GetDesktopHwnd Lib "User" () As Integer

Declare Function GetDesktopWindow Lib "User" () As Integer

' Window field offsets for GetWindowLong() and GetWindowWord()

Global Const GWW_HINSTANCE = (-6)

Global Const GWW_HWNDPARENT = (-8)

Global Const GWW_ID = (-12)

Global Const GWL_WNDPROC = (-4)

Global Const GWL_STYLE = (-16)

Global Const GWL_EXSTYLE = (-20)

'Window Class Functions

Declare Function GetClassWord Lib "User" (ByVal hWnd As Integer, ByVal nIndex As Integer) As Integer

Declare Function SetClassWord Lib "User" (ByVal hWnd As Integer, ByVal nIndex As Integer, ByVal wNewWord As Integer) As Integer

Declare Function GetClassLong Lib "User" (ByVal hWnd As Integer, ByVal nIndex As Integer) As Long

Declare Function SetClassLong Lib "User" (ByVal hWnd As Integer, ByVal nIndex As Integer, ByVal dwNewLong As Long) As Long

Declare Function GetClassInfo Lib "User" (ByVal hInstance As Integer, ByVal lpClassName As String, ByVal lpWndClass As WNDCLASS) As Integer

Declare Function GetClassName Lib "User" (ByVal hWnd As Integer, ByVal lpClassName As String, ByVal nMaxCount As Integer) As Integer

Declare Function UnregisterClass Lib "User" (ByVal lpClassName As String, ByVal hInstance As Integer) As Integer

Declare Function RegisterClass Lib "User" (ByVal lpWndClass As WNDCLASS) As Integer

' Class field offsets for GetClassLong() and GetClassWord()

Global Const GCL_MENUNAME = (-8)

Global Const GCL_WNDPROC = (-24)

Global Const GCW_HBRBACKGROUND = (-10)

Global Const GCW_HCURSOR = (-12)

Global Const GCW_HICON = (-14)

Global Const GCW_HMODULE = (-16)

Global Const GCW_CBWNDEXTRA = (-18)

Global Const GCW_CBCLSEXTRA = (-20)

Global Const GCW_STYLE = (-26)

'Define Windows Processing Functions

Declare Function DefFrameProc Lib "User" (ByVal hWnd As Integer, ByVal hWndMDIClient As Integer, ByVal wParam As Integer, ByVal lParam As Long) As Long

Declare Function DefMDIChildProc Lib "User" (ByVal hWnd As Integer, ByVal wParam As Integer, ByVal lParam As Long) As Long

Declare Function DefWindowProc Lib "User" (ByVal hWnd As Integer, ByVal wParam As Integer, ByVal lParam As Integer, lParam As Any) As Long

'Window Data Structures and Constants

Type WNDCLASS

style As Long
lpfnWndProc As Long
cbClsExtra As Integer
cbWndExtra As Integer
hInstance As Integer
hIcon As Integer
hCursor As Integer
hbrBackground As Integer
lpszMenuName As String
lpszClassName As String

End Type

' Window Styles

Global Const WS_OVERLAPPED = &H00000000&
Global Const WS_POPUP = &H80000000&
Global Const WS_CHILD = &H40000000&
Global Const WS_MINIMIZE = &H20000000&
Global Const WS_VISIBLE = &H10000000&
Global Const WS_DISABLED = &H80000000&
Global Const WS_CLIPSIBLINGS = &H40000000&
Global Const WS_CLIPCHILDREN = &H20000000&
Global Const WS_MAXIMIZE = &H10000000&
Global Const WS_CAPTION = &HC000000& ' WS_BORDER Or WS_DLGFRAME
Global Const WS_BORDER = &H80000000&
Global Const WS_DLGFRAME = &H40000000&
Global Const WS_VSCROLL = &H20000000&
Global Const WS_HSCROLL = &H10000000&
Global Const WS_SYSMENU = &H80000000&
Global Const WS_THICKFRAME = &H40000000&
Global Const WS_GROUP = &H20000000&
Global Const WS_TABSTOP = &H10000000&
Global Const WS_MINIMIZEBOX = &H20000000&
Global Const WS_MAXIMIZEBOX = &H10000000&
Global Const WS_TILED = WS_OVERLAPPED
Global Const WS_ICONIC = WS_MINIMIZE
Global Const WS_SIZEBOX = WS_THICKFRAME

' Common Window Styles

Global Const WS_OVERLAPPEDWINDOW = (WS_OVERLAPPED Or WS_CAPTION Or
WS_SYSMENU Or WS_THICKFRAME Or WS_MINIMIZEBOX Or WS_MAXIMIZEBOX)
Global Const WS_POPUPWINDOW = (WS_POPUP Or WS_BORDER Or WS_SYSMENU)
Global Const WS_CHILDWINDOW = (WS_CHILD)
Global Const WS_TILEDWINDOW = (WS_OVERLAPPEDWINDOW)

' Extended Window Styles

Global Const WS_EX_DLGMODALFRAME = &H00000001&
Global Const WS_EX_NOPARENTNOTIFY = &H00000004&

' Button Control Styles

Global Const BS_PUSHBUTTON = &H00000000&
Global Const BS_DEFPUSHBUTTON = &H00000001&
Global Const BS_CHECKBOX = &H00000002&
Global Const BS_AUTOCHECKBOX = &H00000003&

Global Const BS_RADIOBUTTON = &H4&
Global Const BS_3STATE = &H5&
Global Const BS_AUTO3STATE = &H6&
Global Const BS_GROUPBOX = &H7&
Global Const BS_USERBUTTON = &H8&
Global Const BS_AUTORADIOBUTTON = &H9&
Global Const BS_PUSHBOX = &HA&
Global Const BS_OWNERDRAW = &HB&
Global Const BS_LEFTTEXT = &H20&

' Class styles

Global Const CS_VREDRAW = &H1
Global Const CS_HREDRAW = &H2
Global Const CS_KEYCVTWINDOW = &H4
Global Const CS_DBLCLKS = &H8
Global Const CS_OWNDC = &H20
Global Const CS_CLASSDC = &H40
Global Const CS_PARENTDC = &H80
Global Const CS_NOKEYCVT = &H100
Global Const CS_NOCLOSE = &H200
Global Const CS_SAVEBITS = &H800
Global Const CS_BYTEALIGNCLIENT = &H1000
Global Const CS_BYTEALIGNWINDOW = &H2000
Global Const CS_GLOBALCLASS = &H4000 ' Global window class

' Edit Control Styles

Global Const ES_LEFT = &H0&
Global Const ES_CENTER = &H1&
Global Const ES_RIGHT = &H2&
Global Const ES_MULTILINE = &H4&
Global Const ES_UPPERCASE = &H8&
Global Const ES_LOWERCASE = &H10&
Global Const ES_PASSWORD = &H20&
Global Const ES_AUTOVSCROLL = &H40&
Global Const ES_AUTOHSCROLL = &H80&
Global Const ES_NOHIDESEL = &H100&
Global Const ES_OEMCONVERT = &H400&

Type CREATESTRUCT

lpCreateParams As Long
hInstance As Integer
hMenu As Integer
hwndParent As Integer
cy As Integer
cx As Integer
y As Integer
x As Integer
style As Long
lpzName As Long
lpzClass As Long
ExStyle As Long

End Type

Display and Movement Functions

Window Position Functions

ShowWindow Function

Misc Functions

'Misc Window Display Functions

```
Declare Function IsWindowVisible Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function IsIconic Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function IsZoomed Lib "User" (ByVal hWnd As Integer) As Integer
Declare Sub GetClientRect Lib "User" (ByVal hWnd As Integer, lpRect As RECT)
Declare Sub GetWindowRect Lib "User" (ByVal hWnd As Integer, lpRect As RECT)
Declare Sub ShowOwnedPopups Lib "User" (ByVal hWnd As Integer, ByVal fShow As Integer)
Declare Function OpenIcon Lib "User" (ByVal hWnd As Integer) As Integer
Declare Sub CloseWindow Lib "User" (ByVal hWnd As Integer)
Declare Sub SetWindowText Lib "User" (ByVal hWnd As Integer, ByVal lpString As String)
Declare Function GetWindowText Lib "User" (ByVal hWnd As Integer, ByVal lpString As String, ByVal aint
As Integer) As Integer
Declare Function GetWindowTextLength Lib "User" (ByVal hWnd As Integer) As Integer
```

'ShowWindow Function

Declare Function ShowWindow Lib "User" (ByVal hWnd As Integer, ByVal nCmdShow As Integer) As Integer

' ShowWindow() Commands

Global Const SW_HIDE = 0

Global Const SW_SHOWNORMAL = 1

Global Const SW_NORMAL = 1

Global Const SW_SHOWMINIMIZED = 2

Global Const SW_SHOWMAXIMIZED = 3

Global Const SW_MAXIMIZE = 3

Global Const SW_SHOWNOACTIVATE = 4

Global Const SW_SHOW = 5

Global Const SW_MINIMIZE = 6

Global Const SW_SHOWMINNOACTIVE = 7

Global Const SW_SHOWNA = 8

Global Const SW_RESTORE = 9

' Old ShowWindow() Commands

Global Const HIDE_WINDOW = 0

Global Const SHOW_OPENWINDOW = 1

Global Const SHOW_ICONWINDOW = 2

Global Const SHOW_FULLSCREEN = 3

Global Const SHOW_OPENNOACTIVATE = 4

'Window Position Functions

```
Declare Sub BringWindowToTop Lib "User" (ByVal hWnd As Integer)
Declare Function ArrangeIconicWindows Lib "User" (ByVal hWnd As Integer) As Integer
Declare Sub MoveWindow Lib "User" (ByVal hWnd As Integer, ByVal X As Integer, ByVal Y As Integer,
ByVal nWidth As Integer, ByVal nHeight As Integer, ByVal bRepaint As Integer)
Declare Function BeginDeferWindowPos Lib "User" (ByVal nNumWindows As Integer)
Declare Function DeferWindowPos Lib "User" (ByVal hWinPosInfo As Integer, ByVal hWnd As Integer,
ByVal hWndInsertAfter As Integer, ByVal x, ByVal y, ByVal cx, ByVal cy, ByVal wFlags As Integer)
Declare Sub EndDeferWindowPos Lib "User" (ByVal hWinPosInfo As Integer)
Declare Sub SetWindowPos Lib "User" (ByVal hWnd As Integer, ByVal hWndInsertAfter As Integer, ByVal
X As Integer, ByVal Y As Integer, ByVal cx As Integer, ByVal cy As Integer, ByVal wFlags As Integer)
```

' SetWindowPos Flags

```
Global Const SWP_NOSIZE = &H1
Global Const SWP_NOMOVE = &H2
Global Const SWP_NOZORDER = &H4
Global Const SWP_NOREDRAW = &H8
Global Const SWP_NOACTIVATE = &H10
Global Const SWP_DRAWFRAME = &H20
Global Const SWP_SHOWWINDOW = &H40
Global Const SWP_HIDEWINDOW = &H80
Global Const SWP_NOCOPYBITS = &H100
Global Const SWP_NOREPOSITION = &H200
```

'Input Functions

```
Declare Function EnableWindow Lib "User" (ByVal hWnd As Integer, ByVal aBOOL As Integer) As Integer
Declare Function IsWindowEnabled Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function SetFocusAPI Lib "User" Alias "SetFocus" (ByVal hWnd As Integer) As Integer
Declare Function GetFocus Lib "User" () As Integer
Declare Function GetActiveWindow Lib "User" () As Integer
Declare Function GetCapture Lib "User" () As Integer
Declare Function SetCapture Lib "User" (ByVal hWnd As Integer) As Integer
Declare Sub ReleaseCapture Lib "User" ()
Declare Function GetCurrentTime Lib "User" () As Long
Declare Function GetDoubleClickTime Lib "User" () As Integer
Declare Sub SetDoubleClickTime Lib "User" (ByVal wCount As Integer)
Declare Function GetTickCount Lib "User" () As Long
Declare Function KillTimer Lib "User" (ByVal hWnd As Integer, ByVal nIDEvent As Integer) As Integer
Declare Function SetSysModalWindow Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function SetActiveWindow Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function SwapMouseButton Lib "User" (ByVal bSwap As Integer) As Integer
DeclareFunction SetTimer Lib "User" (ByVal hWnd As Integer, ByVal nIDEvent As Integer, ByVal wElapse
As Long, ByVal lpTimerFunc As FARPROC) As Integer
```

Hardware Functions

These functions are divided into three groups. The constants or structures used by each are included with the functions.

General and Keyboard Functions

Get Device Capabilities

'General and Keyboard Functions

Declare Function EnableHardwareInput Lib "User" (ByVal bEnableInput As Integer) As Integer

Declare Function GetInputState Lib "User" () As Integer

Declare Function GetKeyNameText Lib "Keyboard" (ByVal lParam As Long, ByVal lpBuffer As String, ByVal nSize As Integer) As Integer

Declare Function OemKeyScan Lib "Keyboard" (ByVal wOemChar As Integer) As Long

Declare Function MapVirtualKey Lib "Keyboard" (ByVal wCode As Integer, ByVal wMapType As Integer) As Integer

Declare Function VkKeyScan Lib "Keyboard" (ByVal cChar As Integer) As Integer

Declare Function GetKBCodePage Lib "Keyboard" ()

Declare Function GetKeyboardType Lib "Keyboard" (ByVal nTypeFlag As Integer) As Integer

Declare Function GetKeyState Lib "User" (ByVal nVirtKey As Integer) As Integer

Declare Function GetAsyncKeyState Lib "User" (ByVal vKey As Integer) As Integer

Declare Sub GetKeyboardState Lib "User" (lpKeyState As Any)

Declare Sub SetKeyboardState Lib "User" (lpKeyState As Any)

' Virtual Keys, Standard Set

Global Const VK_LBUTTON = &H1

Global Const VK_RBUTTON = &H2

Global Const VK_CANCEL = &H3

Global Const VK_MBUTTON = &H4 ' NOT contiguous with L & RBUTTON

Global Const VK_BACK = &H8

Global Const VK_TAB = &H9

Global Const VK_CLEAR = &HC

Global Const VK_RETURN = &HD

Global Const VK_SHIFT = &H10

Global Const VK_CONTROL = &H11

Global Const VK_MENU = &H12

Global Const VK_PAUSE = &H13

Global Const VK_CAPITAL = &H14

Global Const VK_ESCAPE = &H1B

Global Const VK_SPACE = &H20

Global Const VK_PRIOR = &H21

Global Const VK_NEXT = &H22

Global Const VK_END = &H23

Global Const VK_HOME = &H24

Global Const VK_LEFT = &H25

Global Const VK_UP = &H26

Global Const VK_RIGHT = &H27

Global Const VK_DOWN = &H28

Global Const VK_SELECT = &H29

Global Const VK_PRINT = &H2A

Global Const VK_EXECUTE = &H2B

Global Const VK_SNAPSHOT = &H2C

'Global Const VK_COPY = &H2C not used by keyboards.

Global Const VK_INSERT = &H2D

Global Const VK_DELETE = &H2E

Global Const VK_HELP = &H2F

' VK_A thru VK_Z are the same as their ASCII equivalents: 'A' thru 'Z'

' VK_0 thru VK_9 are the same as their ASCII equivalents: '0' thru '9'

Global Const VK_NUMPAD0 = &H60

Global Const VK_NUMPAD1 = &H61

Global Const VK_NUMPAD2 = &H62
Global Const VK_NUMPAD3 = &H63
Global Const VK_NUMPAD4 = &H64
Global Const VK_NUMPAD5 = &H65
Global Const VK_NUMPAD6 = &H66
Global Const VK_NUMPAD7 = &H67
Global Const VK_NUMPAD8 = &H68
Global Const VK_NUMPAD9 = &H69
Global Const VK_MULTIPLY = &H6A
Global Const VK_ADD = &H6B
Global Const VK_SEPARATOR = &H6C
Global Const VK_SUBTRACT = &H6D
Global Const VK_DECIMAL = &H6E
Global Const VK_DIVIDE = &H6F
Global Const VK_F1 = &H70
Global Const VK_F2 = &H71
Global Const VK_F3 = &H72
Global Const VK_F4 = &H73
Global Const VK_F5 = &H74
Global Const VK_F6 = &H75
Global Const VK_F7 = &H76
Global Const VK_F8 = &H77
Global Const VK_F9 = &H78
Global Const VK_F10 = &H79
Global Const VK_F11 = &H7A
Global Const VK_F12 = &H7B
Global Const VK_F13 = &H7C
Global Const VK_F14 = &H7D
Global Const VK_F15 = &H7E
Global Const VK_F16 = &H7F

Global Const VK_NUMLOCK = &H90

'Get Device Capabilities

Declare Function GetDeviceCaps Lib "GDI" (ByVal hDC As Integer, ByVal nIndex As Integer) As Integer

' Device Parameters for GetDeviceCaps()

Global Const DRIVERVERSION = 0 ' Device driver version
Global Const TECHNOLOGY = 2 ' Device classification
Global Const HORZSIZE = 4 ' Horizontal size in millimeters
Global Const VERTSIZE = 6 ' Vertical size in millimeters
Global Const HORZRES = 8 ' Horizontal width in pixels
Global Const VERTRES = 10 ' Vertical width in pixels
Global Const BITSPIXEL = 12 ' Number of bits per pixel
Global Const PLANES = 14 ' Number of planes
Global Const NUMBRUSHES = 16 ' Number of brushes the device has
Global Const NUMPENS = 18 ' Number of pens the device has
Global Const NUMMARKERS = 20 ' Number of markers the device has
Global Const NUMFONTS = 22 ' Number of fonts the device has
Global Const NUMCOLORS = 24 ' Number of colors the device supports
Global Const PDEVICESIZE = 26 ' Size required for device descriptor
Global Const CURVECAPS = 28 ' Curve capabilities
Global Const LINECAPS = 30 ' Line capabilities
Global Const POLYGONALCAPS = 32 ' Polygonal capabilities
Global Const TEXTCAPS = 34 ' Text capabilities
Global Const CLIPCAPS = 36 ' Clipping capabilities
Global Const RASTERCAPS = 38 ' Bitblt capabilities
Global Const ASPECTX = 40 ' Length of the X leg
Global Const ASPECTY = 42 ' Length of the Y leg
Global Const ASPECTXY = 44 ' Length of the hypotenuse

Global Const LOGPIXELSX = 88 ' Logical pixels/inch in X
Global Const LOGPIXELSY = 90 ' Logical pixels/inch in Y

Global Const SIZEPALETTE = 104 ' Number of entries in physical palette
Global Const NUMRESERVED = 106 ' Number of reserved entries in palette
Global Const COLORRES = 108 ' Actual color resolution

' Device Capability Masks:

' Device Technologies

Global Const DT_PLOTTER = 0 ' Vector plotter
Global Const DT_RASDISPLAY = 1 ' Raster display
Global Const DT_RASPRINTER = 2 ' Raster printer
Global Const DT_RASCAMERA = 3 ' Raster camera
Global Const DT_CHARSTREAM = 4 ' Character-stream, PLP
Global Const DT_METAFILE = 5 ' Metafile, VDM
Global Const DT_DISPFILE = 6 ' Display-file

' Curve Capabilities

Global Const CC_NONE = 0 ' Curves not supported
Global Const CC_CIRCLES = 1 ' Can do circles
Global Const CC_PIE = 2 ' Can do pie wedges
Global Const CC_CHORD = 4 ' Can do chord arcs
Global Const CC_ELLIPSES = 8 ' Can do ellipese
Global Const CC_WIDE = 16 ' Can do wide lines
Global Const CC_STYLED = 32 ' Can do styled lines

Global Const CC_WIDESTYLED = 64 ' Can do wide styled lines
Global Const CC_INTERIORS = 128 ' Can do interiors

' Line Capabilities

Global Const LC_NONE = 0 ' Lines not supported
Global Const LC_POLYLINE = 2 ' Can do polylines
Global Const LC_MARKER = 4 ' Can do markers
Global Const LC_POLYMARKER = 8 ' Can do polymarkers
Global Const LC_WIDE = 16 ' Can do wide lines
Global Const LC_STYLED = 32 ' Can do styled lines
Global Const LC_WIDESTYLED = 64 ' Can do wide styled lines
Global Const LC_INTERIORS = 128 ' Can do interiors

' Polygonal Capabilities

Global Const PC_NONE = 0 ' Polygons not supported
Global Const PC_POLYGON = 1 ' Can do polygons
Global Const PC_RECTANGLE = 2 ' Can do rectangles
Global Const PC_WINDPOLYGON = 4 ' Can do winding polygons
Global Const PC_TRAPEZOID = 4 ' Can do trapezoids
Global Const PC_SCANLINE = 8 ' Can do scanlines
Global Const PC_WIDE = 16 ' Can do wide borders
Global Const PC_STYLED = 32 ' Can do styled borders
Global Const PC_WIDESTYLED = 64 ' Can do wide styled borders
Global Const PC_INTERIORS = 128 ' Can do interiors

' Polygonal Capabilities

Global Const CP_NONE = 0 ' No clipping of output
Global Const CP_RECTANGLE = 1 ' Output clipped to rects

' Text Capabilities

Global Const TC_OP_CHARACTER = &H1 ' Can do OutputPrecision CHARACTER
Global Const TC_OP_STROKE = &H2 ' Can do OutputPrecision STROKE
Global Const TC_CP_STROKE = &H4 ' Can do ClipPrecision STROKE
Global Const TC_CR_90 = &H8 ' Can do CharRotAbility 90
Global Const TC_CR_ANY = &H10 ' Can do CharRotAbility ANY
Global Const TC_SF_X_YINDEP = &H20 ' Can do ScaleFreedom X_YINDEPENDENT
Global Const TC_SA_DOUBLE = &H40 ' Can do ScaleAbility DOUBLE
Global Const TC_SA_INTEGER = &H80 ' Can do ScaleAbility INTEGER
Global Const TC_SA_CONTIN = &H100 ' Can do ScaleAbility CONTINUOUS
Global Const TC_EA_DOUBLE = &H200 ' Can do EmboldenAbility DOUBLE
Global Const TC_IA_ABLE = &H400 ' Can do ItalisizeAbility ABLE
Global Const TC_UA_ABLE = &H800 ' Can do UnderlineAbility ABLE
Global Const TC_SO_ABLE = &H1000 ' Can do StrikeOutAbility ABLE
Global Const TC_RA_ABLE = &H2000 ' Can do RasterFontAble ABLE
Global Const TC_VA_ABLE = &H4000 ' Can do VectorFontAble ABLE
Global Const TC_RESERVED = &H8000

' Raster Capabilities

Global Const RC_BITBLT = 1 ' Can do standard BLT.
Global Const RC_BANDING = 2 ' Device requires banding support
Global Const RC_SCALING = 4 ' Device requires scaling support
Global Const RC_BITMAP64 = 8 ' Device can support >64K bitmap
Global Const RC_GDI20_OUTPUT = &H10 ' has 2.0 output calls
Global Const RC_DI_BITMAP = &H80 ' supports DIB to memory
Global Const RC_PALETTE = &H100 ' supports a palette
Global Const RC_DIBTODEV = &H200 ' supports DIBitsToDevice

```
Global Const RC_BIGFONT = &H400    ' supports >64K fonts
Global Const RC_STRETCHBLT = &H800  ' supports StretchBlt
Global Const RC_FLOODFILL = &H1000  ' supports FloodFill
Global Const RC_STRETCHDIB = &H2000 ' supports StretchDIBits
```

Painting Functions

General Paint Functions
Region/Rectangle Functions

'General Paint Functions

```
Declare Function BeginPaint Lib "User" (ByVal hWnd As Integer, lpPaint As PAINTSTRUCT) As Integer
Declare Sub EndPaint Lib "User" (ByVal hWnd As Integer, lpPaint As PAINTSTRUCT)
Declare Function DrawText Lib "GDI" (ByVal hdc As Integer, ByVal lpStr As String, ByVal nCount As Integer, lpRect As RECT, ByVal wFormat As Integer) As Integer
Declare Function DrawIcon Lib "GDI" (ByVal hdc As Integer, ByVal X As Integer, ByVal Y As Integer, ByVal hIcon As Integer) As Integer
Declare Function GetWindowDC Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function GetDC Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function ReleaseDC Lib "User" (ByVal hWnd As Integer, ByVal hdc As Integer) As Integer
Declare Sub UpdateWindow Lib "User" (ByVal hWnd As Integer)
Declare Function GrayString Lib "User" (ByVal hdc As Integer, ByVal hBrush As Integer, ByVal lpOutputFunc As FARPROC, ByVal lpData As String, ByVal nCount As Integer, ByVal X As Integer, ByVal Y As Integer, ByVal nWidth As Integer, ByVal nHeight As Integer) As Integer
```

Type PAINTSTRUCT

```
    hdc As Integer
    fErase As Integer
    rcPaint As RECT
    fRestore As Integer
    flncUpdate As Integer
    rgbReserved As String * 16
```

End Type

' Pel Array

Type PELARRAY

```
    paXCount As Integer
    paYCount As Integer
    paXExt As Integer
    paYExt As Integer
    paRGBs As Integer
```

End Type

' DrawText() Format Flags

```
Global Const DT_TOP = &H0
Global Const DT_LEFT = &H0
Global Const DT_CENTER = &H1
Global Const DT_RIGHT = &H2
Global Const DT_VCENTER = &H4
Global Const DT_BOTTOM = &H8
Global Const DT_WORDBREAK = &H10
Global Const DT_SINGLELINE = &H20
Global Const DT_EXPANDTABS = &H40
Global Const DT_TABSTOP = &H80
Global Const DT_NOCLIP = &H100
Global Const DT_EXTERNALLEADING = &H200
Global Const DT_CALCRECT = &H400
Global Const DT_NOPREFIX = &H800
Global Const DT_INTERNAL = &H1000
```

'Region/Rectangle Painting Functions

```
Declare Function GetUpdateRect Lib "User" (ByVal hWnd As Integer, lpRect As RECT, ByVal bErase As Integer) As Integer
Declare Function GetUpdateRgn Lib "User" (ByVal hWnd As Integer, ByVal hRgn As Integer, ByVal fErase As Integer) As Integer
Declare Function ExcludeUpdateRgn Lib "User" (ByVal hDC As Integer, ByVal hWnd As Integer) As Integer
Declare Sub InvalidateRect Lib "User" (ByVal hWnd As Integer, lpRect As RECT, ByVal bErase As Integer)
Declare Sub ValidateRect Lib "User" (ByVal hWnd As Integer, lpRect As RECT)
Declare Sub InvalidateRgn Lib "User" (ByVal hWnd As Integer, ByVal hRgn As Integer, ByVal bErase As Integer)
Declare Sub ValidateRgn Lib "User" (ByVal hWnd As Integer, ByVal hRgn As Integer)
Declare Sub DrawFocusRect Lib "User" (ByVal hDC As Integer, lpRect As RECT)
Declare Function FillRect Lib "User" (ByVal hDC As Integer, lpRect As RECT, ByVal hBrush As Integer) As Integer
Declare Function FrameRect Lib "User" (ByVal hDC As Integer, lpRect As RECT, ByVal hBrush As Integer) As Integer
Declare Sub InvertRect Lib "User" (ByVal hDC As Integer, lpRect As RECT)
```

Dialog Box Functions

General Functions

Item and Button Functions

Data Constants and Structures

'General Dialog Box Functions

```
Declare Sub EndDialog Lib "User" (ByVal hDlg As Integer, ByVal nResult As Integer)
Declare Function GetDlgCtrlID Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function GetDialogBaseUnits Lib "User" () As Long
Declare Function DefDlgProc Lib "User" (ByVal hDlg As Integer, ByVal wParam As Integer, ByVal lParam As Integer, lParam As Any) As Long
Declare Function IsDialogMessage Lib "User" (ByVal hDlg As Integer, lpMsg As MSG) As Integer
Declare Sub MapDialogRect Lib "User" (ByVal hDlg As Integer, lpRect As RECT)
Declare FunctionDlgDirList Lib "User" (ByVal hDlg As Integer, ByVal lpPathSpec As String, ByVal nIDListBox As Integer, ByVal nIDStaticPath As Integer, ByVal wFileType As Integer) As Integer
Declare FunctionDlgDirSelect Lib "User" (ByVal hDlg As Integer, ByVal lpString As String, ByVal nIDListBox As Integer) As Integer
Declare FunctionDlgDirListComboBox Lib "User" (ByVal hDlg As Integer, ByVal lpPathSpec As String, ByVal nIDComboBox As Integer, ByVal nIDStaticPath As Integer, ByVal wFileType As Integer) As Integer
Declare FunctionDlgDirSelectComboBox Lib "User" (ByVal hDlg As Integer, ByVal lpString As String, ByVal nIDComboBox As Integer) As Integer
```

'Dialog Item and Button Functions

```
Declare Function GetDlgItem Lib "User" (ByVal hDlg As Integer, ByVal nIDDlgItem As Integer) As Integer
Declare Sub SetDlgItemInt Lib "User" (ByVal hDlg As Integer, ByVal nIDDlgItem As Integer, ByVal wValue
As Integer, ByVal bSigned As Integer)
Declare Function GetDlgItemInt Lib "User" (ByVal hDlg As Integer, ByVal nIDDlgItem As Integer,
lpTranslated As Integer, ByVal bSigned As Integer) As Integer
Declare Sub SetDlgItemText Lib "User" (ByVal hDlg As Integer, ByVal nIDDlgItem As Integer, ByVal
lpString As String)
Declare Function GetDlgItemText Lib "User" (ByVal hDlg As Integer, ByVal nIDDlgItem As Integer, ByVal
lpString As String, ByVal nMaxCount As Integer) As Integer
Declare Function SendDlgItemMessage Lib "User" (ByVal hDlg As Integer, ByVal nIDDlgItem As Integer,
ByVal wParam As Integer, ByVal lParam As Integer, lParam As Any) As Long
Declare Function GetNextDlgGroupItem Lib "User" (ByVal hDlg As Integer, ByVal hCtl As Integer, ByVal
bPrevious As Integer) As Integer
Declare Function GetNextDlgTabItem Lib "User" (ByVal hDlg As Integer, ByVal hCtl As Integer, ByVal
bPrevious As Integer) As Integer
Declare Sub CheckDlgButton Lib "User" (ByVal hDlg As Integer, ByVal nIDButton As Integer, ByVal
wCheck As Integer)
Declare Sub CheckRadioButton Lib "User" (ByVal hDlg As Integer, ByVal nIDFirstButton As Integer, ByVal
nIDLastButton As Integer, ByVal nIDCheckButton As Integer)
Declare Function IsDlgButtonChecked Lib "User" (ByVal hDlg As Integer, ByVal nIDButton As Integer) As
Integer
```

'Dialog Constants and Structures

' Dialog Manager Routines

Type MDICREATESTRUCT

szClass As Long
szTitle As Long
hOwner As Integer
x As Integer
y As Integer
cx As Integer
cy As Integer
style As Long
lParam As Long

End Type

Global Const CW_USEDEFAULT = &H8000

Type CLIENTCREATESTRUCT

hWindowMenu As Integer
idFirstChild As Integer

End Type

' Dialog Styles

Global Const DS_ABSALIGN = &H1&

Global Const DS_SYSMODAL = &H2&

Global Const DS_LOCALEEDIT = &H20& ' Edit items get Local storage.

Global Const DS_SETFONT = &H40& ' User specified font for Dlg controls

Global Const DS_MODALFRAME = &H80& ' Can be combined with WS_CAPTION

Global Const DS_NOIDLEMSG = &H100& ' WM_ENTERIDLE message will not be sent

Global Const DC_HASDEFID = &H534% '0x534B

' Dialog Codes

Global Const DLGC_WANTARROWS = &H1 ' Control wants arrow keys

Global Const DLGC_WANTTAB = &H2 ' Control wants tab keys

Global Const DLGC_WANTALLKEYS = &H4 ' Control wants all keys

Global Const DLGC_WANTMESSAGE = &H4 ' Pass message to control

Global Const DLGC_HASSETSEL = &H8 ' Understands EM_SETSEL message

Global Const DLGC_DEFPUSHBUTTON = &H10 ' Default pushbutton

Global Const DLGC_UNDEFPUSHBUTTON = &H20 ' Non-default pushbutton

Global Const DLGC_RADIOBUTTON = &H40 ' Radio button

Global Const DLGC_WANTCHARS = &H80 ' Want WM_CHAR messages

Global Const DLGC_STATIC = &H100 ' Static item: don't include

Global Const DLGC_BUTTON = &H2000 ' Button item: can be checked

' Dialog Box Command IDs

Global Const IDOK = 1

Global Const IDCANCEL = 2

Global Const IDABORT = 3

Global Const IDRETRY = 4

Global Const IDIGNORE = 5

Global Const IDYES = 6

Global Const IDNO = 7

Global Const LB_CTLCODE = 0&

' Listbox Return Values

Global Const LB_OKAY = 0

Global Const LB_ERR = (-1)

Global Const LB_ERRSPACE = (-2)

' Listbox Notification Codes

Global Const LBN_ERRSPACE = (-2)

Global Const LBN_SELCHANGE = 1

Global Const LBN_DBLCLK = 2

Global Const LBN_SELCANCEL = 3

Global Const LBN_SETFOCUS = 4

Global Const LBN_KILLFOCUS = 5

' Listbox Styles

Global Const LBS_NOTIFY = &H1&

Global Const LBS_SORT = &H2&

Global Const LBS_NOREDRAW = &H4&

Global Const LBS_MULTIPLESEL = &H8&

Global Const LBS_OWNERDRAWFIXED = &H10&

Global Const LBS_OWNERDRAWVARIABLE = &H20&

Global Const LBS_HASSTRINGS = &H40&

Global Const LBS_USETABSTOPS = &H80&

Global Const LBS_NOINTEGRALHEIGHT = &H100&

Global Const LBS_MULTICOLUMN = &H200&

Global Const LBS_WANTKEYBOARDINPUT = &H400&

Global Const LBS_EXTENDEDSEL = &H800&

Global Const LBS_STANDARD = (LBS_NOTIFY Or LBS_SORT Or WS_VSCROLL Or WS_BORDER)

' Combo Box return Values

Global Const CB_OKAY = 0

Global Const CB_ERR = (-1)

Global Const CB_ERRSPACE = (-2)

' Combo Box Notification Codes

Global Const CBN_ERRSPACE = (-1)

Global Const CBN_SELCHANGE = 1

Global Const CBN_DBLCLK = 2

Global Const CBN_SETFOCUS = 3

Global Const CBN_KILLFOCUS = 4

Global Const CBN_EDITCHANGE = 5

Global Const CBN_EDITUPDATE = 6

Global Const CBN_DROPDOWN = 7

' Combo Box styles

Global Const CBS_SIMPLE = &H1&

Global Const CBS_DROPDOWN = &H2&

Global Const CBS_DROPDOWNLIST = &H3&

Global Const CBS_OWNERDRAWFIXED = &H10&

Global Const CBS_OWNERDRAWVARIABLE = &H20&

Global Const CBS_AUTOHSCROLL = &H40&

Global Const CBS_OEMCONVERT = &H80&

Global Const CBS_SORT = &H100&

Global Const CBS_HASSTRINGS = &H200&

Global Const CBS_NOINTEGRALHEIGHT = &H400&

' Scroll Bar Styles

Global Const SBS_HORZ = &H0&
Global Const SBS_VERT = &H1&
Global Const SBS_TOPALIGN = &H2&
Global Const SBS_LEFTALIGN = &H2&
Global Const SBS_BOTTOMALIGN = &H4&
Global Const SBS_RIGHTALIGN = &H4&
Global Const SBS_SIZEBOXTOPLEFTALIGN = &H2&
Global Const SBS_SIZEBOXBOTTOMRIGHTALIGN = &H4&
Global Const SBS_SIZEBOX = &H8&

' Static Control Constants

Global Const SS_LEFT = &H0&
Global Const SS_CENTER = &H1&
Global Const SS_RIGHT = &H2&
Global Const SS_ICON = &H3&
Global Const SS_BLACKRECT = &H4&
Global Const SS_GRAYRECT = &H5&
Global Const SS_WHITERECT = &H6&
Global Const SS_BLACKFRAME = &H7&
Global Const SS_GRAYFRAME = &H8&
Global Const SS_WHITEFRAME = &H9&
Global Const SS_USERITEM = &HA&
Global Const SS_SIMPLE = &HB&
Global Const SS_LEFTNOWORDWRAP = &HC&
Global Const SS_NOPREFIX = &H80& ' Don't do "&" character translation

Global Const DLGWINDOWEXTRA = 30 ' Window extra bytes needed for private dialog classes

'Scrolling Functions

```
Declare Sub ScrollWindow Lib "User" (ByVal hWnd As Integer, ByVal XAmount As Integer, ByVal YAmount As Integer, lpRect As RECT, lpClipRect As RECT)
Declare Function ScrollDC Lib "User" (ByVal hDC As Integer, ByVal dx As Integer, ByVal dy As Integer, lpRect As RECT, lpClipRect As RECT, ByVal hRgnUpdate As Integer, lpRgnUpdate As RECT) As Integer
Declare Function SetScrollPos Lib "User" (ByVal hWnd As Integer, ByVal nBar As Integer, ByVal nPos As Integer, ByVal bRedraw As Integer) As Integer
Declare Function GetScrollPos Lib "User" (ByVal hWnd As Integer, ByVal nBar As Integer) As Integer
Declare Sub SetScrollRange Lib "User" (ByVal hWnd As Integer, ByVal nBar As Integer, ByVal nMinPos As Integer, ByVal nMaxPos As Integer, ByVal bRedraw As Integer)
Declare Sub GetScrollRange Lib "User" (ByVal hWnd As Integer, ByVal nBar As Integer, lpMinPos As Integer, lpMaxPos As Integer)
Declare Sub ShowScrollBar Lib "User" (ByVal hWnd As Integer, ByVal wBar As Integer, ByVal bShow As Integer)
```

' Scroll Bar Constants

Global Const SB_HORZ = 0

Global Const SB_VERT = 1

Global Const SB_CTL = 2

Global Const SB_BOTH = 3

Menu-Management Functions

Menu Functions

Menu Item Functions

'Menu Functions

```
Declare Function GetSystemMenu Lib "User" (ByVal hWnd As Integer, ByVal bRevert As Integer) As Integer
Declare Function CreateMenu Lib "User" () As Integer
Declare Function CreatePopupMenu Lib "User" () As Integer
Declare Function DestroyMenu Lib "User" (ByVal hMenu As Integer) As Integer
Declare Function GetSubMenu Lib "User" (ByVal hMenu As Integer, ByVal nPos As Integer) As Integer
Declare Function InsertMenu Lib "User" (ByVal hMenu As Integer, ByVal nPosition As Integer, ByVal wFlags As Integer, ByVal wIDNewItem As Integer, ByVal lpNewItem As Any) As Integer
Declare Function AppendMenu Lib "User" (ByVal hMenu As Integer, ByVal wFlags As Integer, ByVal wIDNewItem As Integer, ByVal lpNewItem As Any) As Integer
Declare Function RemoveMenu Lib "User" (ByVal hMenu As Integer, ByVal nPosition As Integer, ByVal wFlags As Integer) As Integer
Declare Function DeleteMenu Lib "User" (ByVal hMenu As Integer, ByVal nPosition As Integer, ByVal wFlags As Integer) As Integer
Declare Function ModifyMenu Lib "User" (ByVal hMenu As Integer, ByVal nPosition As Integer, ByVal wFlags As Integer, ByVal wIDNewItem As Integer, ByVal lpString As Any) As Integer
Declare Function ChangeMenu Lib "User" (ByVal hMenu As Integer, ByVal wID As Integer, ByVal lpszNew As String, ByVal wIDNew As Integer, ByVal wChange As Integer) As Integer
Declare Function GetMenuCheckMarkDimensions Lib "User" () As Long
Declare Function TrackPopupMenu Lib "User" (ByVal hMenu As Integer, ByVal wFlags As Integer, ByVal x As Integer, ByVal y As Integer, ByVal nReserved As Integer, ByVal hWnd As Integer, lpReserved As Any) As Integer
Declare Function LoadMenuIndirect Lib "User" (lpMenuTemplate As MENUITEMTEMPLATE) As Integer
Declare Function GetMenu Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function SetMenu Lib "User" (ByVal hWnd As Integer, ByVal hMenu As Integer) As Integer
Declare Function GetMenuState Lib "User" (ByVal hMenu As Integer, ByVal wID As Integer, ByVal wFlags As Integer) As Integer
Declare Sub DrawMenuBar Lib "User" (ByVal hWnd As Integer)
```

'Menu Item Functions

Declare Function CheckMenuItem Lib "User" (ByVal hMenu As Integer, ByVal wIDCheckItem As Integer, ByVal wCheck As Integer) As Integer
Declare Function EnableMenuItem Lib "User" (ByVal hMenu As Integer, ByVal wIDEnableItem As Integer, ByVal wEnable As Integer) As Integer
Declare Function GetMenuItemID Lib "User" (ByVal hMenu As Integer, ByVal nPos As Integer) As Integer
Declare Function GetMenuItemCount Lib "User" (ByVal hMenu As Integer) As Integer
Declare Function SetMenuItemBitmaps Lib "User" (ByVal hMenu As Integer, ByVal nPosition As Integer, ByVal wFlags As Integer, ByVal hBitmapUnchecked As Integer, ByVal hBitmapChecked As Integer) As Integer
Declare Function HiliteMenuItem Lib "User" (ByVal hWnd As Integer, ByVal hMenu As Integer, ByVal wIDHiliteItem As Integer, ByVal wHilite As Integer) As Integer
Declare Function GetMenuString Lib "User" (ByVal hMenu As Integer, ByVal wIDItem As Integer, ByVal lpString As String, ByVal nMaxCount As Integer, ByVal wFlag As Integer) As Integer

' Menu flags for Add/Check/EnableMenuItem()

Global Const MF_INSERT = &H0

Global Const MF_CHANGE = &H80

Global Const MF_APPEND = &H100

Global Const MF_DELETE = &H200

Global Const MF_REMOVE = &H1000

Global Const MF_BYCOMMAND = &H0

Global Const MF_BYPOSITION = &H400

Global Const MF_SEPARATOR = &H800

Global Const MF_ENABLED = &H0

Global Const MF_GRAYED = &H1

Global Const MF_DISABLED = &H2

Global Const MF_UNCHECKED = &H0

Global Const MF_CHECKED = &H8

Global Const MF_USECHECKBITMAPS = &H200

Global Const MF_STRING = &H0

Global Const MF_BITMAP = &H4

Global Const MF_OWNERDRAW = &H100

Global Const MF_POPUP = &H10

Global Const MF_MENUBARBREAK = &H20

Global Const MF_MENUBREAK = &H40

Global Const MF_UNHILITE = &H0

Global Const MF_HILITE = &H80

Global Const MF_SYSMENU = &H2000

Global Const MF_HELP = &H4000

Global Const MF_MOUSESELECT = &H8000

Global Const MF_END = &H80

' Menu item resource format

Type MENUITEMTEMPLATEHEADER

 versionNumber As Integer

 offset As Integer

End Type

Type MENUITEMTEMPLATE
 mtOption As Integer
 mtID As Integer
 mtString As Long

End Type

'Information Functions

```
Declare Function AnyPopup Lib "User" () As Integer
'Declare Function WindowFromPoint Lib "User" (ByVal Point As Any) As Integer
Declare Function WindowFromPoint Lib "User" (ByVal Y As Integer, ByVal X As Integer) As Integer
Declare Function ChildWindowFromPoint Lib "User" (ByVal hWnd As Integer, ByVal Point As Any) As Integer
Declare Function GetParent Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function SetParent Lib "User" (ByVal hWndChild As Integer, ByVal hWndNewParent As Integer) As Integer
Declare Function FindWindow Lib "User" (ByVal lpClassName As Any, ByVal lpWindowName As Any) As Integer
Declare Function GetTopWindow Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function GetNextWindow Lib "User" (ByVal hWnd As Integer, ByVal wFlag As Integer) As Integer
Declare Function GetWindowTask Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function GetSysModalWindow Lib "User" () As Integer
Declare Function IsWindow Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function IsChild Lib "User" (ByVal hWndParent As Integer, ByVal hWnd As Integer) As Integer
Declare Function EnumChildWindows Lib "User" (ByVal hWndParent As Integer, ByVal lpEnumFunc As FARPROC, ByVal lParam As Double) As Integer
Declare Function EnumTaskWindows Lib "User" (ByVal hTask As Integer, ByVal lpEnumFunc As FARPROC, ByVal lParam As Double) As Integer
Declare Function EnumWindows Lib "User" (ByVal lpEnumFunc As FARPROC, ByVal lParam As Double) As Integer
```

```
Declare Function GetWindow Lib "User" (ByVal hWnd As Integer, ByVal wCmd As Integer) As Integer
```

```
' GetWindow() Constants
Global Const GW_HWNDFIRST = 0
Global Const GW_HWNDLAST = 1
Global Const GW_HWNDNEXT = 2
Global Const GW_HWNDPREV = 3
Global Const GW_OWNER = 4
Global Const GW_CHILD = 5
```

'System Functions

Declare Function GetCurrentTime Lib "User" () As Long

Declare Function GetSystemMetrics Lib "User" (ByVal nIndex As Integer) As Integer

' GetSystemMetrics() codes

Global Const SM_CXSCREEN = 0

Global Const SM_CYSCREEN = 1

Global Const SM_CXVSCROLL = 2

Global Const SM_CXHSCROLL = 3

Global Const SM_CYCAPTION = 4

Global Const SM_CXBORDER = 5

Global Const SM_CYBORDER = 6

Global Const SM_CXDLGFRAME = 7

Global Const SM_CYDLGFRAME = 8

Global Const SM_CYVTHUMB = 9

Global Const SM_CXHTHUMB = 10

Global Const SM_CXICON = 11

Global Const SM_CYICON = 12

Global Const SM_CXCURSOR = 13

Global Const SM_CYCURSOR = 14

Global Const SM_CYMENU = 15

Global Const SM_CXFULLSCREEN = 16

Global Const SM_CYFULLSCREEN = 17

Global Const SM_CYKANJIWINDOW = 18

Global Const SM_MOUSEPRESENT = 19

Global Const SM_CYVSCROLL = 20

Global Const SM_CXHSCROLL = 21

Global Const SM_DEBUG = 22

Global Const SM_SWAPBUTTON = 23

Global Const SM_RESERVED1 = 24

Global Const SM_RESERVED2 = 25

Global Const SM_RESERVED3 = 26

Global Const SM_RESERVED4 = 27

Global Const SM_CXMIN = 28

Global Const SM_CYMIN = 29

Global Const SM_CXSIZE = 30

Global Const SM_CYSIZE = 31

Global Const SM_CXFRAME = 32

Global Const SM_CYFRAME = 33

Global Const SM_CXMINTRACK = 34

Global Const SM_CYMINTRACK = 35

Global Const SM_CMETRICS = 36

Declare Function GetSysColor Lib "User" (ByVal nIndex As Integer) As Long

Declare Sub SetSysColors Lib "User" (ByVal nChanges As Integer, lpSysColor As Integer, lpColorValues As Any)

' Color Types

Global Const CTLCOLOR_MSGBOX = 0

Global Const CTLCOLOR_EDIT = 1

Global Const CTLCOLOR_LISTBOX = 2

Global Const CTLCOLOR_BTN = 3

Global Const CTLCOLOR_DLG = 4

Global Const CTLCOLOR_SCROLLBAR = 5

Global Const CTLCOLOR_STATIC = 6

Global Const CTLCOLOR_MAX = 8 ' three bits max

Global Const COLOR_SCROLLBAR = 0
Global Const COLOR_BACKGROUND = 1
Global Const COLOR_ACTIVECAPTION = 2
Global Const COLOR_INACTIVECAPTION = 3
Global Const COLOR_MENU = 4
Global Const COLOR_WINDOW = 5
Global Const COLOR_WINDOWFRAME = 6
Global Const COLOR_MENUTEXT = 7
Global Const COLOR_WINDOWTEXT = 8
Global Const COLOR_CAPTIONTEXT = 9
Global Const COLOR_ACTIVEBORDER = 10
Global Const COLOR_INACTIVEBORDER = 11
Global Const COLOR_APPWORKSPACE = 12
Global Const COLOR_HIGHLIGHT = 13
Global Const COLOR_HIGHLIGHTTEXT = 14
Global Const COLOR_BTNFACE = 15
Global Const COLOR_BTNSHADOW = 16
Global Const COLOR_GRAYTEXT = 17
Global Const COLOR_BTNTEXT = 18
Global Const COLOR_ENDCOLORS = COLOR_BTNTEXT

'Clipboard Functions

```
Declare Function OpenClipboard Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function CloseClipboard Lib "User" () As Integer
Declare Function GetClipboardOwner Lib "User" () As Integer
Declare Function SetClipboardViewer Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function GetClipboardViewer Lib "User" () As Integer
Declare Function ChangeClipboardChain Lib "User" (ByVal hWnd As Integer, ByVal hWndNext As Integer) As Integer
Declare Function SetClipboardData Lib "User" (ByVal wFormat As Integer, ByVal hMem As Integer) As Integer
Declare Function GetClipboardData Lib "User" (ByVal wFormat As Integer) As Integer
Declare Function RegisterClipboardFormat Lib "User" (ByVal lpString As String) As Integer
Declare Function CountClipboardFormats Lib "User" () As Integer
Declare Function EnumClipboardFormats Lib "User" (ByVal wFormat As Integer) As Integer
Declare Function GetClipboardFormatName Lib "User" (ByVal wFormat As Integer, ByVal lpString As String, ByVal nMaxCount As Integer) As Integer
Declare Function EmptyClipboard Lib "User" () As Integer
Declare Function IsClipboardFormatAvailable Lib "User" (ByVal wFormat As Integer) As Integer
Declare Function GetPriorityClipboardFormat Lib "User" (lpPriorityList As Integer, ByVal nCount As Integer) As Integer
```

' Predefined Clipboard Formats

```
Global Const CF_TEXT = 1
Global Const CF_BITMAP = 2
Global Const CF_METAFILEPICT = 3
Global Const CF_SYLK = 4
Global Const CF_DIF = 5
Global Const CF_TIFF = 6
Global Const CF_OEMTEXT = 7
Global Const CF_DIB = 8
Global Const CF_PALETTE = 9

Global Const CF_OWNERDISPLAY = &H80
Global Const CF_DSPTEXT = &H81
Global Const CF_DSPBITMAP = &H82
Global Const CF_DSPMETAFILEPICT = &H83
```

'Error Functions

Declare Function FlashWindow Lib "User" (ByVal hWnd As Integer, ByVal bInvert As Integer) As Integer

Declare Function MessageBox Lib "User" (ByVal hWnd As Integer, ByVal lpText As String, ByVal lpCaption As String, ByVal wType As Integer) As Integer

Declare Sub MessageBeep Lib "User" (ByVal wType As Integer)

' MessageBox() Flags

Global Const MB_OK = &H0

Global Const MB_OKCANCEL = &H1

Global Const MB_ABORTRETRYIGNORE = &H2

Global Const MB_YESNOCANCEL = &H3

Global Const MB_YESNO = &H4

Global Const MB_RETRYCANCEL = &H5

Global Const MB_ICONHAND = &H10

Global Const MB_ICONQUESTION = &H20

Global Const MB_ICONEXCLAMATION = &H30

Global Const MB_ICONASTERISK = &H40

Global Const MB_ICONINFORMATION = MB_ICONASTERISK

Global Const MB_ICONSTOP = MB_ICONHAND

Global Const MB_DEFBUTTON1 = &H0

Global Const MB_DEFBUTTON2 = &H100

Global Const MB_DEFBUTTON3 = &H200

Global Const MB_APPLMODAL = &H0

Global Const MB_SYSTEMMODAL = &H1000

Global Const MB_TASKMODAL = &H2000

Global Const MB_NOFOCUS = &H8000

Global Const MB_TYPEMASK = &HF

Global Const MB_ICONMASK = &HF0

Global Const MB_DEFMASK = &HF00

Global Const MB_MODEMASK = &H3000

Global Const MB_MISCMASK = &HC000

'Caret Functions

Declare Sub CreateCaret Lib "User" (ByVal hWnd As Integer, ByVal hBitmap As Integer, ByVal nWidth As Integer, ByVal nHeight As Integer)

Declare Function GetCaretBlinkTime Lib "User" () As Integer

Declare Sub SetCaretBlinkTime Lib "User" (ByVal wMSeconds As Integer)

Declare Sub DestroyCaret Lib "User" ()

Declare Sub HideCaret Lib "User" (ByVal hWnd As Integer)

Declare Sub ShowCaret Lib "User" (ByVal hWnd As Integer)

Declare Sub SetCaretPos Lib "User" (ByVal X As Integer, ByVal Y As Integer)

Declare Sub GetCaretPos Lib "User" (lpPoint As POINTAPI)

'Cursor Functions

```
Declare Function ShowCursor Lib "User" (ByVal bShow As Integer) As Integer
Declare Sub SetCursorPos Lib "User" (ByVal X As Integer, ByVal Y As Integer)
Declare Function SetCursor Lib "User" (ByVal hCursor As Integer) As Integer
Declare Sub GetCursorPos Lib "User" (lpPoint As POINTAPI)
Declare Sub ClipCursor Lib "User" (lpRect As Any)
Declare Function LoadCursor Lib "User" (ByVal hInstance As Integer, ByVal lpCursorName As Any) As Integer
Declare Function CreateCursor Lib "User" (ByVal hInstance%, ByVal nXhotspot%, ByVal nYhotspot%,
ByVal nWidth%, ByVal nHeight%, ByVal lpANDbitPlane As Any, ByVal lpXORbitPlane As Any) As Integer
Declare Function DestroyCursor Lib "User" (ByVal hCursor As Integer) As Integer
```

' Standard Cursor IDs

```
Global Const IDC_ARROW = 32512&
Global Const IDC_IBEAM = 32513&
Global Const IDC_WAIT = 32514&
Global Const IDC_CROSS = 32515&
Global Const IDC_UPARROW = 32516&
Global Const IDC_SIZE = 32640&
Global Const IDC_ICON = 32641&
Global Const IDC_SIZENWSE = 32642&
Global Const IDC_SIZENESW = 32643&
Global Const IDC_SIZEWE = 32644&
Global Const IDC_SIZENS = 32645&
```

'Hook Functions

```
Declare Function CallMsgFilter Lib "User" (lpMsg As MSG, ByVal nCode As Integer) As Integer
Declare Function DefHookProc Lib "User" (ByVal code As Integer, wParam As Long, lpfnNextHook As FARPROC) As Integer
Declare Function SetWindowsHook Lib "User" (ByVal nFilterType As Integer, lpFilterFunc As FARPROC) As Integer
Declare Function UnhookWindowsHook Lib "User" (ByVal nHook As Integer, ByVal lpfnHook As FARPROC) As Integer
```

' SetWindowsHook() codes

```
Global Const WH_MSGFILTER = (-1)
Global Const WH_JOURNALRECORD = 0
Global Const WH_JOURNALPLAYBACK = 1
Global Const WH_KEYBOARD = 2
Global Const WH_GETMESSAGE = 3
Global Const WH_CALLWNDPROC = 4
Global Const WH_CBT = 5
Global Const WH_SYSMSGFILTER = 6
Global Const WH_WINDOWMGR = 7
```

' Hook Codes

```
Global Const HC_LPLPFNNEXT = (-2)
Global Const HC_LPFNNEXT = (-1)
Global Const HC_ACTION = 0
Global Const HC_GETNEXT = 1
Global Const HC_SKIP = 2
Global Const HC_NOREM = 3
Global Const HC_NOREMOVE = 3
Global Const HC_SYSMODALON = 4
Global Const HC_SYSMODALOFF = 5
```

' CBT Hook Codes

```
Global Const HCBT_MOVESIZE = 0
Global Const HCBT_MINMAX = 1
Global Const HCBT_QS = 2
```

' WH_MSGFILTER Filter Proc Codes

```
Global Const MSGF_DIALOGBOX = 0
Global Const MSGF_MESSAGEBOX = 1
Global Const MSGF_MENU = 2
Global Const MSGF_MOVE = 3
Global Const MSGF_SIZE = 4
Global Const MSGF_SCROLLBAR = 5
Global Const MSGF_NEXTWINDOW = 6
```

' Window Manager Hook Codes

```
Global Const WC_INIT = 1
Global Const WC_SWP = 2
Global Const WC_DEFWINDOWPROC = 3
Global Const WC_MINMAX = 4
Global Const WC_MOVE = 5
Global Const WC_SIZE = 6
Global Const WC_DRAWCAPTION = 7
```

'Property Functions

Declare Function SetProp Lib "User" (ByVal hWnd As Integer, ByVal lpString As String, ByVal hData As Integer) As Integer

Declare Function GetProp Lib "User" (ByVal hWnd As Integer, ByVal lpString As Any) As Integer

Declare Function RemoveProp Lib "User" (ByVal hWnd As Integer, ByVal lpString As String) As Integer

Declare Function EnumProps Lib "User" (ByVal hWnd As Integer, lpEnumFunc As FARPROC) As Integer

'Rectangle Functions

```
Declare Sub SetRectEmpty Lib "User" (lpRect As RECT)
Declare Function CopyRect Lib "User" (lpDestRect As RECT, lpSourceRect As RECT) As Integer
Declare Sub InflateRect Lib "User" (lpRect As RECT, ByVal X As Integer, ByVal Y As Integer)
Declare Function IntersectRect Lib "User" (lpDestRect As RECT, lpSrc1Rect As RECT, lpSrc2Rect As RECT) As Integer
Declare Function UnionRect Lib "User" (lpDestRect As RECT, lpSrc1Rect As RECT, lpSrc2Rect As RECT) As Integer
Declare Sub OffsetRect Lib "User" (lpRect As RECT, ByVal X As Integer, ByVal Y As Integer)
Declare Function IsRectEmpty Lib "User" (lpRect As RECT) As Integer
Declare Function EqualRect Lib "User" (lpRect1 As RECT, lpRect2 As RECT) As Integer
Declare Function PtInRect Lib "User" (lpRect As RECT, ByVal Point As Any) As Integer
Declare Sub SetRect Lib "User" (lpRect As RECT, ByVal X1 As Integer, ByVal Y1 As Integer, ByVal X2 As Integer, ByVal Y2 As Integer)
```

' RECT Structure is located in General Purpose Structures

'Device Context Functions

Declare Function CreateDC Lib "GDI" (ByVal IpDriverName As String, ByVal IpDeviceName As String, ByVal IpOutput As String, ByVal IpInitData As String) As Integer

Declare Function CreateIC Lib "GDI" (ByVal IpDriverName As String, ByVal IpDeviceName As String, ByVal IpOutput As String, ByVal IpInitData As String) As Integer

Declare Function CreateCompatibleDC Lib "GDI" (ByVal hDC As Integer) As Integer

Declare Function DeleteDC Lib "GDI" (ByVal hDC As Integer) As Integer

Declare Function SaveDC Lib "GDI" (ByVal hDC As Integer) As Integer

Declare Function RestoreDC Lib "GDI" (ByVal hDC As Integer, ByVal nSavedDC As Integer) As Integer

Declare Function GetDCOrg Lib "GDI" (ByVal hDC As Integer) As Long

Drawing Tool Functions

Pen/Brush Functions

Drawing Object Functions

'Pen/Brush Functions

```
Declare Function CreatePen Lib "GDI" (ByVal nPenStyle As Integer, ByVal nWidth As Integer, ByVal crColor As Long) As Integer
Declare Function CreatePenIndirect Lib "GDI" (lpLogPen As LOGPEN) As Integer
Declare Function CreateSolidBrush Lib "GDI" (ByVal crColor As Long) As Integer
Declare Function CreateHatchBrush Lib "GDI" (ByVal nIndex As Integer, ByVal crColor As Long) As Integer
Declare Function SetBrushOrg Lib "GDI" (ByVal hDC As Integer, ByVal X As Integer, ByVal Y As Integer) As Long
Declare Function GetBrushOrg Lib "GDI" (ByVal hDC As Integer) As Long
Declare Function CreatePatternBrush Lib "GDI" (ByVal hBitmap As Integer) As Integer
Declare Function CreateBrushIndirect Lib "GDI" (lpLogBrush As LOGBRUSH) As Integer
Declare Function CreateDIBPatternBrush Lib "GDI" (ByVal hPackedDIB As Integer, ByVal wUsage As Integer) As Integer
```

' Brush Styles

```
Global Const BS_SOLID = 0
Global Const BS_NULL = 1
Global Const BS_HOLLOW = BS_NULL
Global Const BS_HATCHED = 2
Global Const BS_PATTERN = 3
Global Const BS_INDEXED = 4
Global Const BS_DIBPATTERN = 5
```

' Hatch Styles for CreateHatchBrush

```
Global Const HS_HORIZONTAL = 0      ' ----
Global Const HS_VERTICAL = 1        ' ||||
Global Const HS_FDIAGONAL = 2       ' \\\
Global Const HS_BDIAGONAL = 3       ' ///
Global Const HS_CROSS = 4           ' +++++
Global Const HS_DIAGCROSS = 5       ' xxxxx
```

' Pen Styles

```
Global Const PS_SOLID = 0
Global Const PS_DASH = 1            ' -----
Global Const PS_DOT = 2             ' .....
Global Const PS_DASHDOT = 3         ' _.-.-.-_
Global Const PS_DASHDOTDOT = 4     ' _.-.-.-_
Global Const PS_NULL = 5
Global Const PS_INSIDEFRAME = 6
```

' Logical Pen

```
Type LOGPEN
    lpnStyle As Integer
    lpnWidth As POINTAPI
    lpnColor As Long
End Type
```

' Logical Brush (or Pattern)

```
Type LOGBRUSH
    lbStyle As Integer
    lbColor As Long
    lbHatch As Integer
End Type
```

'Drawing Object Functions

```
Declare Function GetStockObject Lib "GDI" (ByVal nIndex As Integer) As Integer
Declare Function GetObject Lib "GDI" (ByVal hObject As Integer, ByVal nCount As Integer, ByVal lpObject
As Long) As Integer
Declare Function DeleteObject Lib "GDI" (ByVal hObject As Integer) As Integer
Declare Function SelectObject Lib "GDI" (ByVal hDC As Integer, ByVal hObject As Integer) As Integer
Declare Function UnrealizeObject Lib "GDI" (ByVal hObject As Integer) As Integer
Declare Function EnumObjects Lib "User" (ByVal hDC As Integer, ByVal nObjectType As Integer, ByVal
lpObjectFunc As FARPROC, ByVal lpData As String) As Integer
```

' Stock Logical Objects

```
Global Const WHITE_BRUSH = 0
Global Const LTGRAY_BRUSH = 1
Global Const GRAY_BRUSH = 2
Global Const DKGRAY_BRUSH = 3
Global Const BLACK_BRUSH = 4
Global Const NULL_BRUSH = 5
Global Const HOLLOW_BRUSH = NULL_BRUSH
Global Const WHITE_PEN = 6
Global Const BLACK_PEN = 7
Global Const NULL_PEN = 8
Global Const OEM_FIXED_FONT = 10
Global Const ANSI_FIXED_FONT = 11
Global Const ANSI_VAR_FONT = 12
Global Const SYSTEM_FONT = 13
Global Const DEVICE_DEFAULT_FONT = 14
Global Const DEFAULT_PALETTE = 15
Global Const SYSTEM_FIXED_FONT = 16
```

' "GDI OBJ" formats do get DeleteObject()'d

```
Global Const CF_GDIOBJFIRST = &H300
Global Const CF_GDIOBJLAST = &H3FF
```

'Color Palette Functions

```
Declare Function CreatePalette Lib "GDI" (lpLogPalette As LOGPALETTE) As Integer
Declare Function SelectPalette Lib "User" (ByVal hDC As Integer, ByVal hPalette As Integer, ByVal
bForceBackground As Integer) As Integer
Declare Function RealizePalette Lib "User" (ByVal hDC As Integer) As Integer
Declare Function UpdateColors Lib "GDI" (ByVal hDC As Integer) As Integer
Declare Sub AnimatePalette Lib "GDI" (ByVal hPalette As Integer, ByVal wStartIndex As Integer, ByVal
wNumEntries As Integer, lpPaletteColors As PALETTEENTRY)
Declare Function SetPaletteEntries Lib "GDI" (ByVal hPalette As Integer, ByVal wStartIndex As Integer,
ByVal wNumEntries As Integer, lpPaletteEntries As PALETTEENTRY) As Integer
Declare Function GetPaletteEntries Lib "GDI" (ByVal hPalette As Integer, ByVal wStartIndex As Integer,
ByVal wNumEntries As Integer, lpPaletteEntries As PALETTEENTRY) As Integer
Declare Function GetNearestPaletteIndex Lib "GDI" (ByVal hPalette As Integer, ByVal crColor As Long)
As Integer
Declare Function ResizePalette Lib "GDI" (ByVal hPalette As Integer, ByVal nNumEntries As Integer) As
Integer
Declare Function GetSystemPaletteEntries Lib "GDI" (ByVal hDC As Integer, ByVal wStartIndex As
Integer, ByVal wNumEntries As Integer, lpPaletteEntries As PALETTEENTRY) As Integer
Declare Function GetSystemPaletteUse Lib "GDI" (ByVal hDC As Integer) As Integer
Declare Function SetSystemPaletteUse Lib "GDI" (ByVal hDC As Integer, ByVal wUsage As Integer) As
Integer
```

```
' constants for Get/SetSystemPaletteUse()
```

```
Global Const SYSPAL_STATIC = 1
```

```
Global Const SYSPAL_NOSTATIC = 2
```

```
' palette entry flags
```

```
Global Const PC_RESERVED = &H1 ' palette index used for animation
```

```
Global Const PC_EXPLICIT = &H2 ' palette index is explicit to device
```

```
Global Const PC_NOCOLLAPSE = &H4 ' do not match color to system palette
```

```
Type PALETTEENTRY
```

```
    peRed As String * 1
```

```
    peGreen As String * 1
```

```
    peBlue As String * 1
```

```
    peFlags As String * 1
```

```
End Type
```

```
Type LOGPALETTE
```

```
    palVersion As Integer
```

```
    palNumEntries As Integer
```

```
    palPalEntry As String * 252 ' Array length is arbitrary; may be changed
```

```
End Type
```

'Drawing Attribute Functions

```
Declare Function SetBkColor Lib "GDI" (ByVal hDC As Integer, ByVal crColor As Long) As Long
Declare Function GetBkColor Lib "GDI" (ByVal hDC As Integer) As Long
Declare Function SetBkMode Lib "GDI" (ByVal hDC As Integer, ByVal nBkMode As Integer) As Integer
Declare Function GetBkMode Lib "GDI" (ByVal hDC As Integer) As Integer
Declare Function SetTextColor Lib "GDI" (ByVal hDC As Integer, ByVal crColor As Long) As Long
Declare Function GetTextColor Lib "GDI" (ByVal hDC As Integer) As Long
Declare Function SetROP2 Lib "GDI" (ByVal hDC As Integer, ByVal nDrawMode As Integer) As Integer
Declare Function GetROP2 Lib "GDI" (ByVal hDC As Integer) As Integer
Declare Function SetStretchBltMode Lib "GDI" (ByVal hDC As Integer, ByVal nStretchMode As Integer)
As Integer
Declare Function GetStretchBltMode Lib "GDI" (ByVal hDC As Integer) As Integer
Declare Function SetPolyFillMode Lib "GDI" (ByVal hDC As Integer, ByVal nPolyFillMode As Integer) As
Integer
Declare Function GetPolyFillMode Lib "GDI" (ByVal hDC As Integer) As Integer
```

' Binary raster ops

```
Global Const R2_BLACK = 1 ' 0
Global Const R2_NOTMERGEPEN = 2 ' DPon
Global Const R2_MASKNOTPEN = 3 ' DPna
Global Const R2_NOTCOPYPEN = 4 ' PN
Global Const R2_MASKPENNOT = 5 ' PDna
Global Const R2_NOT = 6 ' Dn
Global Const R2_XORPEN = 7 ' DPx
Global Const R2_NOTMASKPEN = 8 ' DPan
Global Const R2_MASKPEN = 9 ' DPa
Global Const R2_NOTXORPEN = 10 ' DPxn
Global Const R2_NOP = 11 ' D
Global Const R2_MERGENOTPEN = 12' DPno
Global Const R2_COPYPEN = 13 ' P
Global Const R2_MERGEPEENNOT = 14' PDno
Global Const R2_MERGEPEEN = 15 ' DPo
Global Const R2_WHITE = 16 ' 1
```

' Ternary raster operations

```
Global Const SRCCOPY = &HCC0020 ' (DWORD) dest = source
Global Const SRCPAINT = &HEE0086 ' (DWORD) dest = source OR dest
Global Const SRCAND = &H8800C6 ' (DWORD) dest = source AND dest
Global Const SRCINVERT = &H660046 ' (DWORD) dest = source XOR dest
Global Const SRCERASE = &H440328 ' (DWORD) dest = source AND (NOT dest )
Global Const NOTSRCCOPY = &H330008 ' (DWORD) dest = (NOT source)
Global Const NOTSRCERASE = &H1100A6 ' (DWORD) dest = (NOT src) AND (NOT dest)
Global Const MERGECOPY = &HC000CA ' (DWORD) dest = (source AND pattern)
Global Const MERGEPAIN = &HBB0226 ' (DWORD) dest = (NOT source) OR dest
Global Const PATCOPY = &HF00021 ' (DWORD) dest = pattern
Global Const PATPAINT = &HFB0A09 ' (DWORD) dest = DPSnoo
Global Const PATINVERT = &H5A0049 ' (DWORD) dest = pattern XOR dest
Global Const DSTINVERT = &H550009 ' (DWORD) dest = (NOT dest)
Global Const BLACKNESS = &H42& ' (DWORD) dest = BLACK
Global Const WHITENESS = &HFF0062' (DWORD) dest = WHITE
```

' StretchBlt() Modes

```
Global Const BLACKONWHITE = 1
Global Const WHITEONBLACK = 2
Global Const COLORONCOLOR = 3
```

' PolyFill() Modes

Global Const ALTERNATE = 1

Global Const WINDING = 2

'Mapping Functions

```
Declare Function SetMapMode Lib "GDI" (ByVal hDC As Integer, ByVal nMapMode As Integer) As Integer
Declare Function GetMapMode Lib "GDI" (ByVal hDC As Integer) As Integer
Declare Function SetWindowOrg Lib "GDI" (ByVal hDC As Integer, ByVal X As Integer, ByVal Y As Integer) As Long
Declare Function GetWindowOrg Lib "GDI" (ByVal hDC As Integer) As Long
Declare Function SetWindowExt Lib "GDI" (ByVal hDC As Integer, ByVal X As Integer, ByVal Y As Integer) As Long
Declare Function GetWindowExt Lib "GDI" (ByVal hDC As Integer) As Long
Declare Function SetViewportOrg Lib "GDI" (ByVal hDC As Integer, ByVal X As Integer, ByVal Y As Integer) As Long
Declare Function GetViewportOrg Lib "GDI" (ByVal hDC As Integer) As Long
Declare Function SetViewportExt Lib "GDI" (ByVal hDC As Integer, ByVal X As Integer, ByVal Y As Integer) As Long
Declare Function GetViewportExt Lib "GDI" (ByVal hDC As Integer) As Long
Declare Function OffsetViewportOrg Lib "GDI" (ByVal hDC As Integer, ByVal X As Integer, ByVal Y As Integer) As Long
Declare Function ScaleViewportExt Lib "GDI" (ByVal hDC As Integer, ByVal Xnum As Integer, ByVal Xdenom As Integer, ByVal Ynum As Integer, ByVal Ydenom As Integer) As Long
Declare Function OffsetWindowOrg Lib "GDI" (ByVal hDC As Integer, ByVal X As Integer, ByVal Y As Integer) As Long
Declare Function ScaleWindowExt Lib "GDI" (ByVal hDC As Integer, ByVal Xnum As Integer, ByVal Xdenom As Integer, ByVal Ynum As Integer, ByVal Ydenom As Integer) As Long
```

'Coordinate Functions

Declare Function ChildWindowFromPoint Lib "User" (ByVal hWnd As Integer, ByVal Point As Any) As Integer

Declare Sub ClientToScreen Lib "User" (ByVal hWnd As Integer, lpPoint As POINTAPI)

Declare Sub ScreenToClient Lib "User" (ByVal hWnd As Integer, lpPoint As POINTAPI)

Declare Function DPToLP Lib "GDI" (ByVal hDC As Integer, lpPoints As POINTAPI, ByVal nCount As Integer) As Integer

Declare Function LPtoDP Lib "GDI" (ByVal hDC As Integer, lpPoints As POINTAPI, ByVal nCount As Integer) As Integer

Declare Function WindowFromPoint Lib "User" (ByVal Point As Any) As Integer

' Coordinate Modes

Global Const ABSOLUTE = 1

Global Const RELATIVE = 2

'Region Functions

```
Declare Function CombineRgn Lib "GDI" (ByVal hDestRgn As Integer, ByVal hSrcRgn1 As Integer, ByVal hSrcRgn2 As Integer, ByVal nCombineMode As Integer) As Integer
Declare Function EqualRgn Lib "GDI" (ByVal hSrcRgn1 As Integer, ByVal hSrcRgn2 As Integer) As Integer
Declare Function OffsetRgn Lib "GDI" (ByVal hRgn As Integer, ByVal X As Integer, ByVal Y As Integer) As Integer
Declare Function GetRgnBox Lib "GDI" (ByVal hRgn As Integer, lpRect As RECT) As Integer
Declare Function CreateRectRgn Lib "GDI" (ByVal X1 As Integer, ByVal Y1 As Integer, ByVal X2 As Integer, ByVal Y2 As Integer) As Integer
Declare Sub SetRectRgn Lib "GDI" (ByVal hRgn As Integer, ByVal X1 As Integer, ByVal Y1 As Integer, ByVal X2 As Integer, ByVal Y2 As Integer)
Declare Function CreateRectRgnIndirect Lib "GDI" (lpRect As RECT) As Integer
Declare Function CreateEllipticRgnIndirect Lib "GDI" (lpRect As RECT) As Integer
Declare Function CreateEllipticRgn Lib "GDI" (ByVal X1 As Integer, ByVal Y1 As Integer, ByVal X2 As Integer, ByVal Y2 As Integer) As Integer
Declare Function CreatePolygonRgn Lib "GDI" (lpPoints As POINTAPI, ByVal nCount As Integer, ByVal nPolyFillMode As Integer) As Integer
Declare Function CreatePolyPolygonRgn Lib "GDI" (lpPoints As POINTAPI, lpPolyCounts As Integer, ByVal nCount As Integer, ByVal nPolyFillMode As Integer) As Integer
Declare Function CreateRoundRectRgn Lib "GDI" (ByVal X1 As Integer, ByVal Y1 As Integer, ByVal X2 As Integer, ByVal Y2 As Integer, ByVal X3 As Integer, ByVal Y3 As Integer) As Integer
Declare Function FillRgn Lib "User" (ByVal hDC As Integer, ByVal hRgn As Integer, ByVal hBrush As Integer) As Integer
Declare Function FrameRgn Lib "User" (ByVal hDC As Integer, ByVal hRgn As Integer, ByVal hBrush As Integer, ByVal nWidth As Integer, ByVal nHeight As Integer) As Integer
Declare Function InvertRgn Lib "User" (ByVal hDC As Integer, ByVal hRgn As Integer) As Integer
Declare Function PaintRgn Lib "User" (ByVal hDC As Integer, ByVal hRgn As Integer) As Integer
Declare Function PtInRegion Lib "User" (ByVal hRgn As Integer, ByVal X As Integer, ByVal Y As Integer) As Integer
Declare Function RectInRegion Lib "User" (ByVal hRgn As Integer, lpRect As RECT) As Integer
```

' Region Flags

```
Global Const ERRORAPI = 0
Global Const NULLREGION = 1
Global Const SIMPLEREGION = 2
Global Const COMPLEXREGION = 3
```

' CombineRgn() Styles

```
Global Const RGN_AND = 1
Global Const RGN_OR = 2
Global Const RGN_XOR = 3
Global Const RGN_DIFF = 4
Global Const RGN_COPY = 5
```

'Clipping Functions

```
Declare Function GetClipBox Lib "GDI" (ByVal hDC As Integer, lpRect As RECT) As Integer
Declare Function IntersectClipRect Lib "GDI" (ByVal hDC As Integer, ByVal X1 As Integer, ByVal Y1 As Integer, ByVal X2 As Integer, ByVal Y2 As Integer) As Integer
Declare Function OffsetClipRgn Lib "GDI" (ByVal hDC As Integer, ByVal X As Integer, ByVal Y As Integer) As Integer
Declare Function ExcludeClipRect Lib "GDI" (ByVal hDC As Integer, ByVal X1 As Integer, ByVal Y1 As Integer, ByVal X2 As Integer, ByVal Y2 As Integer) As Integer
Declare Function PtVisible Lib "GDI" (ByVal hDC As Integer, ByVal X As Integer, ByVal Y As Integer) As Integer
Declare Function RectVisible Lib "User" (ByVal hDC As Integer, lpRect As RECT) As Integer
Declare Function SelectClipRgn Lib "GDI" (ByVal hDC As Integer, ByVal hRgn As Integer) As Integer
```

'Line Output Functions

Declare Function Arc Lib "GDI" (ByVal hDC As Integer, ByVal X1 As Integer, ByVal Y1 As Integer, ByVal X2 As Integer, ByVal Y2 As Integer, ByVal X3 As Integer, ByVal Y3 As Integer, ByVal X4 As Integer, ByVal Y4 As Integer) As Integer

Declare Function LineTo Lib "GDI" (ByVal hDC As Integer, ByVal X As Integer, ByVal Y As Integer) As Integer

Declare Function Polyline Lib "GDI" (ByVal hDC As Integer, lpPoints As POINTAPI, ByVal nCount As Integer) As Integer

Declare Function MoveTo Lib "GDI" (ByVal hDC As Integer, ByVal X As Integer, ByVal Y As Integer) As Long

Declare Sub LineDDA Lib "GDI" (ByVal X1 As Integer, ByVal Y1 As Integer, ByVal X2 As Integer, ByVal Y2 As Integer, ByVal lpLineFunc As FARPROC, ByVal lpData As String)

'Ellipse and Polygon Functions

Declare Function Ellipse Lib "GDI" (ByVal hDC As Integer, ByVal X1 As Integer, ByVal Y1 As Integer, ByVal X2 As Integer, ByVal Y2 As Integer) As Integer

Declare Function Chord Lib "GDI" (ByVal hDC As Integer, ByVal X1 As Integer, ByVal Y1 As Integer, ByVal X2 As Integer, ByVal Y2 As Integer, ByVal X3 As Integer, ByVal Y3 As Integer, ByVal X4 As Integer, ByVal Y4 As Integer) As Integer

Declare Function Pie Lib "GDI" (ByVal hDC As Integer, ByVal X1 As Integer, ByVal Y1 As Integer, ByVal X2 As Integer, ByVal Y2 As Integer, ByVal X3 As Integer, ByVal Y3 As Integer, ByVal X4 As Integer, ByVal Y4 As Integer) As Integer

Declare Sub DrawFocusRect Lib "User" (ByVal hDC As Integer, lpRect As RECT)

Declare Function Polygon Lib "GDI" (ByVal hDC As Integer, lpPoints As POINTAPI, ByVal nCount As Integer) As Integer

Declare Function PolyPolygon Lib "GDI" (ByVal hDC As Integer, lpPoints As POINTAPI, lpPolyCounts As Integer, ByVal nCount As Integer) As Integer

Declare Function Rectangle Lib "GDI" (ByVal hDC As Integer, ByVal X1 As Integer, ByVal Y1 As Integer, ByVal X2 As Integer, ByVal Y2 As Integer) As Integer

Declare Function RoundRect Lib "GDI" (ByVal hDC As Integer, ByVal X1 As Integer, ByVal Y1 As Integer, ByVal X2 As Integer, ByVal Y2 As Integer, ByVal X3 As Integer, ByVal Y3 As Integer) As Integer

Bitmap-Management Functions

Blit Functions

Bitmap Functions

Pixel and FloodFill Functions

'Bitmap Blt Functions

Declare Function PatBlt Lib "GDI" (ByVal hDC As Integer, ByVal X As Integer, ByVal Y As Integer, ByVal nWidth As Integer, ByVal nHeight As Integer, ByVal dwRop As Long) As Integer

Declare Function BitBlt Lib "GDI" (ByVal hDestDC As Integer, ByVal X As Integer, ByVal Y As Integer, ByVal nWidth As Integer, ByVal nHeight As Integer, ByVal hSrcDC As Integer, ByVal XSrc As Integer, ByVal YSrc As Integer, ByVal dwRop As Long) As Integer

Declare Function StretchBlt% Lib "GDI" (ByVal hDC%, ByVal X%, ByVal Y%, ByVal nWidth%, ByVal nHeight%, ByVal hSrcDC%, ByVal XSrc%, ByVal YSrc%, ByVal nSrcWidth%, ByVal nSrcHeight%, ByVal dwRop%)

'Bitmap Functions

```
Declare Function CreateBitmap Lib "GDI" (ByVal nWidth As Integer, ByVal nHeight As Integer, ByVal nPlanes As Integer, ByVal nBitCount As Integer, ByVal lpBits As Any) As Integer
Declare Function CreateBitmapIndirect Lib "GDI" (lpBitmap As BITMAP) As Integer
Declare Function CreateCompatibleBitmap Lib "GDI" (ByVal hDC As Integer, ByVal nWidth As Integer, ByVal nHeight As Integer) As Integer
Declare Function CreateDiscardableBitmap Lib "GDI" (ByVal hDC As Integer, ByVal nWidth As Integer, ByVal nHeight As Integer) As Integer
Declare Function SetBitmapBits Lib "GDI" (ByVal hBitmap As Integer, ByVal dwCount As Long, ByVal lpBits As Any) As Long
Declare Function GetBitmapBits Lib "GDI" (ByVal hBitmap As Integer, ByVal dwCount As Long, ByVal lpBits As Any) As Long
Declare Function SetBitmapDimension Lib "GDI" (ByVal hBitmap As Integer, ByVal X As Integer, ByVal Y As Integer) As Long
Declare Function GetBitmapDimension Lib "GDI" (ByVal hBitmap As Integer) As Long
Declare Function LoadBitmap Lib "User" (ByVal hInstance As Integer, ByVal lpBitmapName As Any) As Integer
Declare Function CreateDIBitmap Lib "GDI" (ByVal hDC As Integer, lpInfoHeader As BITMAPINFOHEADER, ByVal dwUsage As Long, ByVal lpInitBits As String, lpInitInfo As BITMAPINFO, ByVal wUsage As Integer) As Integer
Declare Function StretchDIBits# Lib "GDI" (ByVal hDC#, ByVal X#, ByVal Y#, ByVal dX#, ByVal dY#, ByVal SrcX#, ByVal SrcY#, ByVal wSrcWidth#, ByVal wSrcHeight#, ByVal lpBits As String, lpBitsInfo As BITMAPINFO, ByVal wUsage#, ByVal dwRop#)
Declare Function SetDIBits Lib "GDI" (ByVal aHDC As Integer, ByVal hBitmap As Integer, ByVal nStartScan As Integer, ByVal nNumScans As Integer, ByVal lpBits As String, lpBI As BITMAPINFO, ByVal wUsage As Integer) As Integer
Declare Function GetDIBits Lib "GDI" (ByVal aHDC As Integer, ByVal hBitmap As Integer, ByVal nStartScan As Integer, ByVal nNumScans As Integer, ByVal lpBits As String, lpBI As BITMAPINFO, ByVal wUsage As Integer) As Integer
Declare Function SetDIBitsToDevice% Lib "GDI" (ByVal hDC#, ByVal X#, ByVal Y#, ByVal dX#, ByVal dY#, ByVal SrcX#, ByVal SrcY#, ByVal Scan#, ByVal NumScans#, ByVal Bits As String, BitsInfo As BITMAPINFO, ByVal wUsage#)

' constants for CreateDIBitmap
' DIB color table identifiers
Global Const DIB_RGB_COLORS = 0 ' color table in RGBTriples
Global Const DIB_PAL_COLORS = 1 ' color table in palette indices

Global Const CBM_INIT = &H4& ' initialize bitmap
```

'Pixel and Flood Fill Functions

Declare Function SetPixel Lib "GDI" (ByVal hdc As Integer, ByVal X As Integer, ByVal Y As Integer, ByVal crColor As Long) As Long

Declare Function GetPixel Lib "GDI" (ByVal hdc As Integer, ByVal X As Integer, ByVal Y As Integer) As Long

Declare Function FloodFill Lib "GDI" (ByVal hdc As Integer, ByVal X As Integer, ByVal Y As Integer, ByVal crColor As Long) As Integer

Declare Function ExtFloodFill Lib "GDI" (ByVal hdc As Integer, ByVal X As Integer, ByVal Y As Integer, ByVal crColor As Long, ByVal wFillType As Integer) As Integer

' ExtFloodFill style flags

Global Const FLOODFILLBORDER = 0

Global Const FLOODFILLSURFACE = 1

'Text Functions

Declare Function ExtTextOut Lib "GDI" (ByVal hDC As Integer, ByVal X As Integer, ByVal Y As Integer, ByVal wOptions As Integer, lpRect As Any, ByVal lpString As String, ByVal nCount As Integer, lpDx As Any) As Integer

Declare Function TextOut Lib "GDI" (ByVal hDC As Integer, ByVal X As Integer, ByVal Y As Integer, ByVal lpString As String, ByVal nCount As Integer) As Integer

Declare Function TabbedTextOut Lib "GDI" (ByVal hDC As Integer, ByVal X As Integer, ByVal Y As Integer, ByVal lpString As String, ByVal nCount As Integer, ByVal nTabPositions As Integer, lpnTabStopPositions As Integer, ByVal nTabOrigin As Integer) As Long

Declare Function SetTextJustification Lib "GDI" (ByVal hDC As Integer, ByVal nBreakExtra As Integer, ByVal nBreakCount As Integer) As Integer

Declare Function GetTextExtent Lib "GDI" (ByVal hDC As Integer, ByVal lpString As String, ByVal nCount As Integer) As Long

Declare Function GetTabbedTextExtent Lib "GDI" (ByVal hDC As Integer, ByVal lpString As String, ByVal nCount As Integer, ByVal nTabPositions As Integer, lpnTabStopPositions As Integer) As Long

Declare Function GetTextFace Lib "GDI" (ByVal hDC As Integer, ByVal nCount As Integer, ByVal lpFacename As String) As Integer

Declare Function GetTextMetrics Lib "GDI" (ByVal hDC As Integer, lpMetrics As TEXTMETRIC) As Integer

Declare Function SetTextAlign Lib "GDI" (ByVal hDC As Integer, ByVal wFlags As Integer) As Integer

Declare Function GetTextAlign Lib "GDI" (ByVal hDC As Integer) As Integer

Global Const ETO_GRAYED = 1

Global Const ETO_OPAQUE = 2

Global Const ETO_CLIPPED = 4

Type TEXTMETRIC

tmHeight As Integer

tmAscent As Integer

tmDescent As Integer

tmInternalLeading As Integer

tmExternalLeading As Integer

tmAveCharWidth As Integer

tmMaxCharWidth As Integer

tmWeight As Integer

tmItalic As String * 1

tmUnderlined As String * 1

tmStruckOut As String * 1

tmFirstChar As String * 1

tmLastChar As String * 1

tmDefaultChar As String * 1

tmBreakChar As String * 1

tmPitchAndFamily As String * 1

tmCharSet As String * 1

tmOverhang As Integer

tmDigitizedAspectX As Integer

tmDigitizedAspectY As Integer

End Type

'Font Functions

```
Declare Function AddFontResource Lib "GDI" (ByVal lpFilename As Any) As Integer
Declare Function RemoveFontResource Lib "GDI" (ByVal lpFilename As Any) As Integer
Declare Function GetCharWidth Lib "GDI" (ByVal hDC As Integer, ByVal wFirstChar As Integer, ByVal wLastChar As Integer, lpBuffer As Integer) As Integer
Declare Function CreateFont% Lib "GDI" (ByVal H%, ByVal W%, ByVal E%, ByVal O%, ByVal W%, ByVal I%, ByVal U%, ByVal S%, ByVal C%, ByVal OP%, ByVal CP%, ByVal Q%, ByVal PAF%, ByVal F$)
Declare Function CreateFontIndirect Lib "GDI" (lpLogFont As LOGFONT) As Integer
Declare Function SetMapperFlags Lib "GDI" (ByVal hDC As Integer, ByVal dwFlag As Long) As Long
Declare Function GetCurrentPosition Lib "GDI" (ByVal hDC As Integer) As Long
Declare Function GetAspectRatioFilter Lib "GDI" (ByVal hDC As Integer) As Long
Declare Function GetNearestColor Lib "GDI" (ByVal hDC As Integer, ByVal crColor As Long) As Long
Declare Function SetTextCharacterExtra Lib "GDI" (ByVal hDC As Integer, ByVal nCharExtra As Integer) As Integer
Declare Function GetTextCharacterExtra Lib "GDI" (ByVal hDC As Integer) As Integer
Declare Function EnumFonts Lib "GDI" (ByVal hDC As Integer, ByVal lpFacename As String, ByVal lpFontFunc As FARPROC, ByVal lpData As String) As Integer
```

' Logical Font

```
Global Const LF_FACESIZE = 32
```

Type LOGFONT

```
    lfHeight As Integer
    lfWidth As Integer
    lfEscapement As Integer
    lfOrientation As Integer
    lfWeight As Integer
    lfItalic As String * 1
    lfUnderline As String * 1
    lfStrikeOut As String * 1
    lfCharSet As String * 1
    lfOutPrecision As String * 1
    lfClipPrecision As String * 1
    lfQuality As String * 1
    lfPitchAndFamily As String * 1
    lfFaceName As String * LF_FACESIZE
```

End Type

```
Global Const OUT_DEFAULT_PRECIS = 0
Global Const OUT_STRING_PRECIS = 1
Global Const OUT_CHARACTER_PRECIS = 2
Global Const OUT_STROKE_PRECIS = 3
```

```
Global Const CLIP_DEFAULT_PRECIS = 0
Global Const CLIP_CHARACTER_PRECIS = 1
Global Const CLIP_STROKE_PRECIS = 2
```

```
Global Const DEFAULT_QUALITY = 0
Global Const DRAFT_QUALITY = 1
Global Const PROOF_QUALITY = 2
```

```
Global Const DEFAULT_PITCH = 0
Global Const FIXED_PITCH = 1
Global Const VARIABLE_PITCH = 2
```

Global Const ANSI_CHARSET = 0
Global Const SYMBOL_CHARSET = 2
Global Const SHIFTJIS_CHARSET = 128
Global Const OEM_CHARSET = 255

' Font Families

Global Const FF_DONTCARE = 0 ' Don't care or don't know.
Global Const FF_ROMAN = 16 ' Variable stroke width, serified.

' Times Roman, Century Schoolbook, etc.
Global Const FF_SWISS = 32 ' Variable stroke width, sans-serifed.

' Helvetica, Swiss, etc.
Global Const FF_MODERN = 48 ' Constant stroke width, serified or sans-serifed.

' Pica, Elite, Courier, etc.
Global Const FF_SCRIPT = 64 ' Cursive, etc.
Global Const FF_DECORATIVE = 80 ' Old English, etc.

' Font Weights

Global Const FW_DONTCARE = 0
Global Const FW_THIN = 100
Global Const FW_EXTRALIGHT = 200
Global Const FW_LIGHT = 300
Global Const FW_NORMAL = 400
Global Const FW_MEDIUM = 500
Global Const FW_SEMIBOLD = 600
Global Const FW_BOLD = 700
Global Const FW_EXTRABOLD = 800
Global Const FW_HEAVY = 900

Global Const FW_ULTRALIGHT = FW_EXTRALIGHT
Global Const FW_REGULAR = FW_NORMAL
Global Const FW_DEMIBOLD = FW_SEMIBOLD
Global Const FW_ULTRABOLD = FW_EXTRABOLD
Global Const FW_BLACK = FW_HEAVY

' Background Modes

Global Const TRANSPARENT = 1
Global Const OPAQUE = 2

' Mapping Modes

Global Const MM_TEXT = 1
Global Const MM_LOMETRIC = 2
Global Const MM_HIMETRIC = 3
Global Const MM_LOENGLISH = 4
Global Const MM_HIENGLISH = 5
Global Const MM_TWIPS = 6
Global Const MM_ISOTROPIC = 7
Global Const MM_ANISOTROPIC = 8

'Metafile Functions

```
Declare Function CreateMetaFile Lib "GDI" (lpString As Any) As Integer
Declare Function CloseMetaFile Lib "GDI" (ByVal hMF As Integer) As Integer
Declare Function GetMetaFileBits Lib "GDI" (ByVal hMF As Integer) As Integer
Declare Function SetMetaFileBits Lib "GDI" (ByVal hMem As Integer) As Integer
Declare Function GetMetaFile Lib "GDI" (ByVal lpFilename As String) As Integer
Declare Function DeleteMetaFile Lib "GDI" (ByVal hMF As Integer) As Integer
Declare Function CopyMetaFile Lib "GDI" (ByVal hMF As Integer, ByVal lpFilename As String) As Integer
Declare Function PlayMetaFile Lib "GDI" (ByVal hDC As Integer, ByVal hMF As Integer) As Integer
Declare Sub PlayMetaFileRecord Lib "GDI" (ByVal hDC As Integer, lpHandletable As Integer,
lpMetaRecord As METARECORD, ByVal nHandles As Integer)
Declare Function EnumMetaFile Lib "GDI" (ByVal hDC As Integer, ByVal hMF As Integer, ByVal
lpCallbackFunc As FARPROC, ByVal lpClientData As String) As Integer
```

' Metafile Functions

```
Global Const META_SETBKCOLOR = &H201
Global Const META_SETBKMODE = &H102
Global Const META_SETMAPMODE = &H103
Global Const META_SETROP2 = &H104
Global Const META_SETRELABS = &H105
Global Const META_SETPOLYFILLMODE = &H106
Global Const META_SETSTRETCHBLTMODE = &H107
Global Const META_SETTEXTCHAREXTRA = &H108
Global Const META_SETTEXTCOLOR = &H209
Global Const META_SETTEXTJUSTIFICATION = &H20A
Global Const META_SETWINDOWORG = &H20B
Global Const META_SETWINDOWEXT = &H20C
Global Const META_SETVIEWPORTORG = &H20D
Global Const META_SETVIEWPORTEXT = &H20E
Global Const META_OFFSETWINDOWORG = &H20F
Global Const META_SCALEWINDOWEXT = &H400
Global Const META_OFFSETVIEWPORTORG = &H211
Global Const META_SCALEVIEWPORTEXT = &H412
Global Const META_LINETO = &H213
Global Const META_MOVETO = &H214
Global Const META_EXCLUDECLIPRECT = &H415
Global Const META_INTERSECTCLIPRECT = &H416
Global Const META_ARC = &H817
Global Const META_ELLIPSE = &H418
Global Const META_FLOODFILL = &H419
Global Const META_PIE = &H81A
Global Const META_RECTANGLE = &H41B
Global Const META_ROUNDRECT = &H61C
Global Const META_PATBLT = &H61D
Global Const META_SAVEDC = &H1E
Global Const META_SETPIXEL = &H41F
Global Const META_OFFSETCLIPRGN = &H220
Global Const META_TEXTOUT = &H521
Global Const META_BITBLT = &H922
Global Const META_STRETCHBLT = &HB23
Global Const META_POLYGON = &H324
Global Const META_POLYLINE = &H325
Global Const META_ESCAPE = &H626
Global Const META_RESTOREDC = &H127
Global Const META_FILLREGION = &H228
```

Global Const META_FRAMEREGION = &H429
Global Const META_INVERTREGION = &H12A
Global Const META_PAINTREGION = &H12B
Global Const META_SELECTCLIPREGION = &H12C
Global Const META_SELECTOBJECT = &H12D
Global Const META_SETTEXTALIGN = &H12E
Global Const META_DRAWTEXT = &H62F

Global Const META_CHORD = &H830
Global Const META_SETMAPPERFLAGS = &H231
Global Const META_EXTTEXTOUT = &Ha32
Global Const META_SETDIBTODEV = &Hd33
Global Const META_SELECTPALETTE = &H234
Global Const META_REALIZEPALETTE = &H35
Global Const META_ANIMATEPALETTE = &H436
Global Const META_SETPALENTRIES = &H37
Global Const META_POLYPOLYGON = &H538
Global Const META_RESIZEPALETTE = &H139

Global Const META_DIBBITBLT = &H940
Global Const META_DIBSTRETCHBLT = &Hb41
Global Const META_DIBCREATEPATTERNBRUSH = &H142
Global Const META_STRETCHDIB = &Hf43

Global Const META_DELETEOBJECT = &H1f0

Global Const META_CREATEPALETTE = &Hf7
Global Const META_CREATEBRUSH = &HF8
Global Const META_CREATEPATTERNBRUSH = &H1F9
Global Const META_CREATEPENINDIRECT = &H2FA
Global Const META_CREATEFONTINDIRECT = &H2FB
Global Const META_CREATEBRUSHINDIRECT = &H2FC
Global Const META_CREATEBITMAPINDIRECT = &H2FD
Global Const META_CREATEBITMAP = &H6FE
Global Const META_CREATEREGION = &H6FF

Type METARECORD
 rdSize As Long
 rdFunction As Integer
 rdParm As String * 512 ' Array length is arbitrary; may be changed
End Type

Type METAFILEPICT
 mm As Integer
 xExt As Integer
 yExt As Integer
 hMF As Integer
End Type

Type METAHEADER
 mtType As Integer
 mtHeaderSize As Integer
 mtVersion As Integer
 mtSize As Long
 mtNoObjects As Integer
 mtMaxRecord As Long

mtNoParameters As Integer
End Type

'Printer Control Functions

Declare Function DeviceCapabilities Lib "GDI" (ByVal lpDeviceName As String, ByVal lpPort As String, ByVal nIndex As Long, ByVal lpOutput As String, ByVal lpDevMode As DEVMODE) As Double

Declare Sub DeviceMode Lib "GDI" (ByVal hWnd As Integer, ByVal hModule As Integer, ByVal lpDeviceName As String, ByVal lpOutput As String)

Declare Function ExtDeviceMode (ByVal hWnd As Integer, ByVal hDriver As Integer, ByVal lpDevModeOutput As DEVMODE, ByVal lpDeviceName As String, ByVal lpPort As String, ByVal lpDevModeInput As DEVMODE, ByVal lpProfile As String, ByVal wMode As Long) As Integer

' **The following definitions are for information only. The values are not known.**

' DeviceCapabilities nIndex values: (all values unknown)

Global Const DC_BINNAMES = 0

Global Const DC_BINS = 0

Global Const DC_DRIVER = 0

Global Const DC_DUPLEX = 0

Global Const DC_EXTRA = 0

Global Const DC_FIELDS = 0

Global Const DC_MAXEXTENT = 0

Global Const DC_MINEXTENT = 0

Global Const DC_PAPERS = 0

Global Const DC_PAPERSIZE = 0

Global Const DC_SIZE = 0

Global Const DC_VERSION = 0

Type DEVMODE

dmDeviceName As String*32

dmSpecVersion As Long

dmDriverVersion As Long

dmSize As Long

dmDriverExtra As Long

dmFields As Double

dmOrientation As Integer

dmPaperSize As Integer

dmPaperLength As Integer

dmPaperWidth As Integer

dmScale As Integer

dmCopies As Integer

dmDefaultSource As Integer

dmPrintQuality As Integer

dmColor As Integer

dmDuplex As Integer

dmDriverData As String

End Type

Global Const DMORIENT_PORTRAIT = 1

Global Const DMORIENT_LANDSCAPE = 2

Global Const DMPAPER_LETTER = 1

Global Const DMPAPER_LEGAL = 2

Global Const DMPAPER_A4 = 3

Global Const DMPAPER_CSHEET = 4

Global Const DMPAPER_DSHEET = 5

Global Const DMPAPER_ESHEET = 6

Global Const DMPAPER_ENV_9 = 9

Global Const DMPAPER_ENV_10 = 10
Global Const DMPAPER_ENV_11 = 11
Global Const DMPAPER_ENV_12 = 12
Global Const DMPAPER_ENV_14 = 14

Global Const DMBIN_DEFAULT = 0
Global Const DMBIN_UPPER = 1
Global Const DMBIN_LOWER = 2
Global Const DMBIN_MANUAL = 3
Global Const DMBIN_TRACTOR = 4
Global Const DMBIN_ENVELOPE = 5

Global Const DMRES_HIGH = (-4)
Global Const DMRES_MEDIUM = (-3)
Global Const DMRES_LOW = (-2)
Global Const DMRES_DRAFT = (-1)

Global Const DMCOLOR_COLOR = (-1)
Global Const DMCOLOR_MONOCHROME = (-2)

Global Const DMDUP_SIMPLEX = 1
Global Const DMDUP_HORIZONTAL = 2
Global Const DMDUP_VERTICAL = 3

'Printer Escape Functions

Declare Function Escape Lib "GDI" (ByVal hDC As Integer, ByVal nEscape As Integer, ByVal nCount As Integer, lpInData As Any, lpOutData As Any) As Integer

' GDI Escapes

Global Const NEWFRAME = 1
Global Const ABORTDOC = 2
Global Const NEXTBAND = 3
Global Const SETCOLORTABLE = 4
Global Const GETCOLORTABLE = 5
Global Const FLUSHOUTPUT = 6
Global Const DRAFTMODE = 7
Global Const QUERYESCSUPPORT = 8
Global Const SETABORTPROC = 9
Global Const STARTDOC = 10
Global Const ENDDOCAPI = 11
Global Const GETPHYSPAGESIZE = 12
Global Const GETPRINTINGOFFSET = 13
Global Const GETSCALINGFACTOR = 14
Global Const MFCOMMENT = 15
Global Const GETPENWIDTH = 16
Global Const SETCOPYCOUNT = 17
Global Const SELECTPAPERSOURCE = 18
Global Const DEVICEDATA = 19
Global Const PASSTHROUGH = 19
Global Const GETTECHNOLGY = 20
Global Const GETTECHNOLOGY = 20
Global Const SETENDCAP = 21
Global Const SETLINEJOIN = 22
Global Const SETMITERLIMIT = 23
Global Const BANDINFO = 24
Global Const DRAWPATTERNRECT = 25
Global Const GETVECTORPENSIZE = 26
Global Const GETVECTORBRUSHSIZE = 27
Global Const ENABLEDUPLEX = 28
Global Const GETSETPAPERBINS = 29
Global Const GETSETPRINTORIENT = 30
Global Const ENUMPAPERBINS = 31
Global Const SETDIBSCALING = 32
Global Const EPSPRINTING = 33
Global Const ENUMPAPERMETRICS = 34
Global Const GETSETPAPERMETRICS = 35
Global Const POSTSCRIPT_DATA = 37
Global Const POSTSCRIPT_IGNORE = 38
Global Const GETEXTENDEDTEXTMETRICS = 256
Global Const GETTEXTTABLE = 257
Global Const GETPAIRKERNTABLE = 258
Global Const GETTRACKKERNTABLE = 259
Global Const EXTTEXTOUT = 512
Global Const ENABLERELATIEWIDTHS = 768
Global Const ENABLEPAIRKERNING = 769
Global Const SETKERTRACK = 770
Global Const SETALLJUSTVALUES = 771
Global Const SETCHARSET = 772

Global Const STRETCHBLT = 2048
Global Const BEGIN_PATH = 4096
Global Const CLIP_TO_PATH = 4097
Global Const END_PATH = 4098
Global Const EXT_DEVICE_CAPS = 4099
Global Const RESTORE_CTM = 4100
Global Const SAVE_CTM = 4101
Global Const SET_ARC_DIRECTION = 4102
Global Const SET_BACKGROUND_COLOR = 4103
Global Const SET_POLY_MODE = 4104
Global Const SET_SCREEN_ANGLE = 4105
Global Const SET_SPREAD = 4106
Global Const TRANSFORM_CTM = 4107
Global Const SET_CLIP_BOX = 4108
Global Const SET_BOUNDS = 4109
Global Const SET_MIRROR_MODE = 4110
' Spooler Error Codes

Global Const SP_NOTREPORTED = &H4000
Global Const SP_ERROR = (-1)
Global Const SP_APPABORT = (-2)
Global Const SP_USERABORT = (-3)
Global Const SP_OUTOFDISK = (-4)
Global Const SP_OUTOFMEMORY = (-5)

Global Const PR_JOBSTATUS = &H0

'Environment Functions

Declare Function SetEnvironment Lib "GDI" (ByVal lpPortName As String, ByVal lpEnviron As String, ByVal nCount As Integer) As Integer

Declare Function GetEnvironment Lib "GDI" (ByVal lpPortName As String, lpEnviron As Any, ByVal nMaxCount As Integer) As Integer

'Module Management Functions

```
Declare Function GetVersion Lib "Kernel" () As Integer
Declare Function GetModuleHandle Lib "Kernel" (ByVal lpModuleName As String) As Integer
Declare Function GetModuleUsage Lib "Kernel" (ByVal hModule As Integer) As Integer
Declare Function GetModuleFileName Lib "Kernel" (ByVal hModule As Integer, ByVal lpFilename As String, ByVal nSize As Integer) As Integer
Declare Function GetInstanceData Lib "Kernel" (ByVal hInstance As Integer, ByVal pData As Integer, ByVal nCount As Integer) As Integer
Declare Function LoadLibrary Lib "Kernel" (ByVal lpLibFileName As String) As Integer
Declare Function LoadModule Lib "Kernel" (ByVal lpModuleName As String, lpParameterBlock As PARAMETERBLOCK) As Integer
Declare Sub FreeModule Lib "Kernel" (ByVal hModule As Integer)
Declare Sub FreeLibrary Lib "Kernel" (ByVal hLibModule As Integer)
Declare Sub FreeProcInstance Lib "Kernel" (ByVal lpProc As FARPROC)
Declare Function GetCodeHandle Lib "Kernel" (ByVal lpProc As FARPROC) As Integer
Declare Function GetProcAddress Lib "Kernel" (ByVal hModule As Integer, ByVal lpProcName As String) As FARPROC
Declare Function MakeProcInstance Lib "Kernel" (ByVal lpProc As FARPROC, ByVal hInstance As Integer) As FARPROC

' ParameterBlock description structure for use with LoadModule
Type PARAMETERBLOCK
    wEnvSeg As Integer
    lpCmdLine As Long
    lpCmdShow As Long
    dwReserved As Long
End Type
```

Memory Management Functions

System -Functions

Global Functions

Local Functions

'Memory-Management System Functions

```
Declare Function GetFreeSpace Lib "Kernel" (ByVal wFlags As Integer) As Long
Declare Function GetWinFlags Lib "Kernel" () As Long
Declare Sub LimitEmsPages Lib "Kernel" (ByVal dwKbytes As Long)
Declare Function LockSegment Lib "Kernel" (ByVal wSegment As Integer) As Integer
Declare Function UnlockSegment Lib "Kernel" (ByVal wSegment As Integer) As Integer
Declare Function SetSwapAreaSize Lib "Kernel" (ByVal rsSize As Integer) As Long
Declare Sub SwitchStackBack Lib "Kernel" ()
Declare Sub SwitchStackTo Lib "Kernel" (ByVal wStackSegment As Integer, ByVal wStackPointer As Integer, ByVal wStackTop As Integer)
Declare Function DefineHandleTable Lib "Kernel" (ByVal wOffset As Long) As Integer
Declare Function LockData Lib "Kernel" (ByVal Dummy As Integer) As Integer
Declare Function UnlockData Lib "Kernel" (ByVal Dummy As Integer) As Integer
```

' GetWinFlags Values

```
Global Const WF_PMODE = &H1
Global Const WF_CPU286 = &H2
Global Const WF_CPU386 = &H4
Global Const WF_CPU486 = &H8
Global Const WF_STANDARD = &H10
Global Const WF_WIN286 = &H10
Global Const WF_ENHANCED = &H20
Global Const WF_WIN386 = &H20
Global Const WF_CPU086 = &H40
Global Const WF_CPU186 = &H80
Global Const WF_LARGEFRAME = &H100
Global Const WF_SMALLFRAME = &H200
Global Const WF_80x87 = &H400
```

'Memory-Management Global Functions

```
Declare Function GlobalAlloc Lib "Kernel" (ByVal wFlags As Integer, ByVal dwBytes As Long) As Integer
Declare Function GlobalCompact Lib "Kernel" (ByVal dwMinFree As Long) As Long
Declare Function GlobalFree Lib "Kernel" (ByVal hMem As Integer) As Integer
Declare Function GlobalHandle Lib "Kernel" (ByVal wMem As Integer) As Long
Declare Function GlobalLock Lib "Kernel" (ByVal hMem As Integer) As Long
Declare Function GlobalReAlloc Lib "Kernel" (ByVal hMem As Integer, ByVal dwBytes As Long, ByVal wFlags As Integer) As Integer
```

'NOTE: instead of declaring the function GlobalDiscard and calling
' GlobalDiscard(hMem), call GlobalReAlloc(hMem, 0, GMEM_MOVEABLE)

```
Declare Function GlobalSize Lib "Kernel" (ByVal hMem As Integer) As Long
Declare Function GlobalUnlock Lib "Kernel" (ByVal hMem As Integer) As Integer
Declare Function GlobalFlags Lib "Kernel" (ByVal hMem As Integer) As Integer
Declare Function GlobalWire Lib "Kernel" (ByVal hMem As Integer) As Long
Declare Function GlobalUnWire Lib "Kernel" (ByVal hMem As Integer) As Integer
Declare Function GlobalUnlock Lib "Kernel" (ByVal hMem As Integer) As Integer
Declare Function GlobalLRUNewest Lib "Kernel" (ByVal hMem As Integer) As Integer
Declare Function GlobalLRUOldest Lib "Kernel" (ByVal hMem As Integer) As Integer
Declare Function GlobalDOSAlloc Lib "Kernel" (ByVal dwBytes As Double) As FARPROC
Declare Function GlobalDOSFree Lib "Kernel" (ByVal wSelector As Long) As Long
Declare Sub GlobalNotify Lib "Kernel" (ByVal lpNotifyProc As FARPROC)
```

' Global Memory Flags

```
Global Const GMEM_FIXED = &H0
Global Const GMEM_MOVEABLE = &H2
Global Const GMEM_NOCOMPACT = &H10
Global Const GMEM_NODISCARD = &H20
Global Const GMEM_ZEROINIT = &H40
Global Const GMEM_MODIFY = &H80
Global Const GMEM_DISCARDABLE = &H100
Global Const GMEM_NOT_BANKED = &H1000
Global Const GMEM_SHARE = &H2000
Global Const GMEM_DDESHARE = &H2000
Global Const GMEM_NOTIFY = &H4000
Global Const GMEM_LOWER = GMEM_NOT_BANKED
```

' Flags returned by GlobalFlags (in addition to GMEM_DISCARDABLE)

```
Global Const GMEM_DISCARDED = &H4000
Global Const GMEM_LOCKCOUNT = &HFF
```

```
Global Const GHND = (GMEM_MOVEABLE Or GMEM_ZEROINIT)
```

```
Global Const GPTR = (GMEM_FIXED Or GMEM_ZEROINIT)
```

' "Private" formats don't get GlobalFree()'d

```
Global Const CF_PRIVATEFIRST = &H200
```

```
Global Const CF_PRIVATELAST = &H2FF
```

'Memory-Management Local Functions

```
Declare Function LocalAlloc Lib "Kernel" (ByVal wFlags As Integer, ByVal wBytes As Integer) As Integer
Declare Function LocalCompact Lib "Kernel" (ByVal wMinFree As Integer) As Integer
Declare Function LocalFree Lib "Kernel" (ByVal hMem As Integer) As Integer
Declare Function LocalHandle Lib "Kernel" (ByVal wMem As Integer) As Integer
Declare Function LocalInit Lib "Kernel" (ByVal wSegment As Integer, ByVal pStart As Integer, ByVal pEnd
As Integer) As Integer
Declare Function LocalLock Lib "Kernel" (ByVal hMem As Integer) As Integer '(returns a near pointer)
Declare Function LocalReAlloc Lib "Kernel" (ByVal hMem As Integer, ByVal wBytes As Integer, ByVal
wFlags As Integer) As Integer
```

'NOTE: instead of declaring the function LocalDiscard and calling
' LocalDiscard(hMem), call LocalReAlloc(hMem, 0, LMEM_MOVEABLE)

```
Declare Function LocalSize Lib "Kernel" (ByVal hMem As Integer) As Integer
Declare Function LocalUnlock Lib "Kernel" (ByVal hMem As Integer) As Integer
Declare Function LocalFlags Lib "Kernel" (ByVal hMem As Integer) As Integer
Declare Function LocalShrink Lib "Kernel" (ByVal hSeg As Integer, ByVal wSize As Integer) As Integer
```

' Flags returned by LocalFlags (in addition to LMEM_DISCARDABLE)
Global Const LMEM_DISCARDED = &H4000
Global Const LMEM_LOCKCOUNT = &HFF

' Local Memory Flags
Global Const LMEM_FIXED = &H0
Global Const LMEM_MOVEABLE = &H2
Global Const LMEM_NOCOMPACT = &H10
Global Const LMEM_NODISCARD = &H20
Global Const LMEM_ZEROINIT = &H40
Global Const LMEM_MODIFY = &H80
Global Const LMEM_DISCARDABLE = &HF00

Global Const LHND = (LMEM_MOVEABLE+LMEM_ZEROINIT)
Global Const LPTR = (LMEM_FIXED+LMEM_ZEROINIT)

Global Const NONZEROLHND = (LMEM_MOVEABLE)
Global Const NONZEROLPTR = (LMEM_FIXED)

Global Const LNOTIFY_OUTOFMEM = 0
Global Const LNOTIFY_MOVE = 1
Global Const LNOTIFY_DISCARD = 2

'Segment Functions

```
Declare Function AllocDStoCSAlias Lib "Kernel" (ByVal wSelector As Integer) As Integer
Declare Function AllocSelector Lib "Kernel" (ByVal wSelector As Integer) As Integer
Declare Function ChangeSelector Lib "Kernel" (ByVal wDestSelector As Integer, ByVal wSourceSelector
As Integer) As Integer
Declare Function FreeSelector Lib "Kernel" (ByVal wSelector As Integer) As Integer
Declare Function GlobalPageLock Lib "Kernel" (ByVal wSelector As Integer) As Integer
Declare Function GlobalPageUnlock Lib "Kernel" (ByVal wSelector As Integer) As Integer
Declare Sub GlobalFix Lib "Kernel" (ByVal hMem As Integer)
Declare Function GlobalUnfix Lib "Kernel" (ByVal hMem As Integer) As Integer

Declare Function LockSegment Lib "Kernel" (ByVal wSegment As Integer) As Integer
Declare Function UnlockSegment Lib "Kernel" (ByVal wSegment As Integer) As Integer
Declare Function DefineHandleTable Lib "Kernel" (ByVal wOffset As Long) As Integer
Declare Sub GetCodeInfo Lib "Kernel" (ByVal lpProc As FARPROC, ByVal lpSegInfo As Any)
```

Operating System Interrupt Functions

- ' DOS3Call and NetBIOSCall can only be used in Assembly routines
- ' They are located in KERNEL.EXE

'Task Functions

'NOTE: Catch/Throw expect a long pointer to an 18-byte buffer (lpCatchBuf)

' eg:

```
'      Dim Buffer(1 To 9) As Integer  
'      result% = Catch (Buffer(1))
```

```
Declare Function Catch Lib "Kernel" (lpCatchBuf As Any) As Integer  
Declare Sub Throw Lib "Kernel" (lpCatchBuf As Any, ByVal nThrowBack As Integer)  
Declare Function ExitWindows Lib "User" (ByVal dwReserved As Long, ByVal wReturnCode) As Integer  
Declare Function GetCurrentPDB Lib "Kernel" () As Integer  
Declare Sub Yield Lib "Kernel" ()  
Declare Function GetCurrentTask Lib "Kernel" () As Integer  
Declare Function GetDOSEnvironment Lib "Kernel" () As Long  
Declare Function GetNumTasks Lib "Kernel" () As Integer  
Declare Function SetErrorMode Lib "Kernel" (ByVal wMode As Integer) As Integer
```

' WEP fSystemExit flag values

```
Global Const WEP_SYSTEM_EXIT = 1
```

```
Global Const WEP_FREE_DLL = 0
```

'Resource Management Functions

Declare Function FindResource Lib "Kernel" (ByVal hInstance As Integer, ByVal lpName As String, ByVal lpType As Any) As Integer
Declare Function LoadResource Lib "Kernel" (ByVal hInstance As Integer, ByVal hResInfo As Integer) As Integer
Declare Function FreeResource Lib "Kernel" (ByVal hResData As Integer) As Integer
Declare Function LockResource Lib "Kernel" (ByVal hResData As Integer) As Long
Declare Function AllocResource Lib "Kernel" (ByVal hInstance As Integer, ByVal hResInfo As Integer, ByVal dwSize As Long) As Integer
Declare Function SizeofResource Lib "Kernel" (ByVal hInstance As Integer, ByVal hResInfo As Integer) As Integer
Declare Function AccessResource Lib "Kernel" (ByVal hInstance As Integer, ByVal hResInfo As Integer) As Integer
Declare Function LoadAccelerators Lib "User" (ByVal hInstance As Integer, ByVal lpTableName As String) As Integer
Declare Function LoadBitmap Lib "User" (ByVal hInstance As Integer, ByVal lpBitmapName As Any) As Integer
Declare Function LoadCursor Lib "User" (ByVal hInstance As Integer, ByVal lpCursorName As Any) As Integer
Declare Function LoadIcon Lib "User" (ByVal hInstance As Integer, ByVal lpIconName As Any) As Integer
Declare Function LoadMenu Lib "User" (ByVal hInstance As Integer, ByVal lpString As String) As Integer
Declare Function LoadString Lib "User" (ByVal hInstance As Integer, ByVal wID As Integer, ByVal lpBuffer As Any, ByVal nBufferMax As Integer) As Integer
Declare Function UnlockResource Lib "Kernel" Alias "GlobalUnlock" (ByVal hMem As Integer) As Integer
Declare Function SetResourceHandler Lib "Kernel" (ByVal hInstance As Integer, ByVal lpType As String, ByVal lpLoadFunc As FARPROC) As FARPROC
Declare Function CreateIcon Lib "User" (ByVal hInstance%, ByVal nWidth%, ByVal nHeight%, ByVal nPlanes%, ByVal nBitsPixel%, ByVal lpANDbits As Any, ByVal lpXORbits As Any) As Integer
Declare Function DestroyIcon Lib "User" (ByVal hIcon As Integer) As Integer

' Standard Icon IDs

Global Const IDI_APPLICATION = 32512&
Global Const IDI_HAND = 32513&
Global Const IDI_QUESTION = 32514&
Global Const IDI_EXCLAMATION = 32515&
Global Const IDI_ASTERISK = 32516&

' Predefined Resource Types

Global Const RT_CURSOR = 1&
Global Const RT_BITMAP = 2&
Global Const RT_ICON = 3&
Global Const RT_MENU = 4&
Global Const RT_DIALOG = 5&
Global Const RT_STRING = 6&
Global Const RT_FONTDIR = 7&
Global Const RT_FONT = 8&
Global Const RT_ACCELERATOR = 9&
Global Const RT_RCDATA = 10&

' OEM Resource Ordinal Numbers

Global Const OBM_CLOSE = 32754
Global Const OBM_UPARROW = 32753
Global Const OBM_DNARROW = 32752
Global Const OBM_RGARROW = 32751

Global Const OBM_LFARROW = 32750
Global Const OBM_REDUCE = 32749
Global Const OBM_ZOOM = 32748
Global Const OBM_RESTORE = 32747
Global Const OBM_REDUCED = 32746
Global Const OBM_ZOOMD = 32745
Global Const OBM_RESTORED = 32744
Global Const OBM_UPARROWD = 32743
Global Const OBM_DNARROWD = 32742
Global Const OBM_RGARROWD = 32741
Global Const OBM_LFARROWD = 32740
Global Const OBM_MNARROW = 32739
Global Const OBM_COMBO = 32738

Global Const OBM_OLD_CLOSE = 32767
Global Const OBM_SIZE = 32766
Global Const OBM_OLD_UPARROW = 32765
Global Const OBM_OLD_DNARROW = 32764
Global Const OBM_OLD_RGARROW = 32763
Global Const OBM_OLD_LFARROW = 32762
Global Const OBM_BTFSIZE = 32761
Global Const OBM_CHECK = 32760
Global Const OBM_CHECKBOXES = 32759
Global Const OBM_BTNCORNERS = 32758
Global Const OBM_OLD_REDUCE = 32757
Global Const OBM_OLD_ZOOM = 32756
Global Const OBM_OLD_RESTORE = 32755

Global Const OCR_NORMAL = 32512
Global Const OCR_IBEAM = 32513
Global Const OCR_WAIT = 32514
Global Const OCR_CROSS = 32515
Global Const OCR_UP = 32516
Global Const OCR_SIZE = 32640
Global Const OCR_ICON = 32641
Global Const OCR_SIZENWSE = 32642
Global Const OCR_SIZENESW = 32643
Global Const OCR_SIZEWE = 32644
Global Const OCR_SIZENS = 32645
Global Const OCR_SIZEALL = 32646
Global Const OCR_ICOCUR = 32647

Global Const OIC_SAMPLE = 32512
Global Const OIC_HAND = 32513
Global Const OIC_QUES = 32514
Global Const OIC_BANG = 32515
Global Const OIC_NOTE = 32516

'String Manipulation Functions

```
Declare Function AnsiToOem Lib "Keyboard" (ByVal lpAnsiStr As String, ByVal lpOemStr As String) As Integer
Declare Function OemToAnsi Lib "Keyboard" (ByVal lpOemStr As String, ByVal lpAnsiStr As String) As Integer
Declare Sub AnsiToOemBuff Lib "Keyboard" (ByVal lpAnsiStr As String, ByVal lpOemStr As String, ByVal nLength As Integer)
Declare Sub OemToAnsiBuff Lib "Keyboard" (ByVal lpOemStr As String, ByVal lpAnsiStr As String, ByVal nLength As Integer)
Declare Function AnsiUpper Lib "User" (ByVal lpString As String) As String
Declare Function AnsiUpperBuff Lib "User" (ByVal lpString As String, ByVal aWORD As Integer) As Integer
Declare Function AnsiLower Lib "User" (ByVal lpString As String) As Long
Declare Function AnsiLowerBuff Lib "User" (ByVal lpString As String, ByVal aWORD As Integer) As Integer
Declare Function AnsiNext Lib "User" (ByVal lpString As String) As Long
Declare Function AnsiPrev Lib "User" (ByVal lpString As String, ByVal lpString As String) As Long
Declare Function IsCharAlpha Lib "User" (ByVal cChar As Integer) As Integer
Declare Function IsCharAlphaNumeric Lib "User" (ByVal cChar As Integer) As Integer
Declare Function IsCharUpper Lib "User" (ByVal cChar As Integer) As Integer
Declare Function IsCharLower Lib "User" (ByVal cChar As Integer) As Integer
Declare Function Istrcmp Lib "User" (ByVal lpString1 As Any, ByVal lpString2 As Any) As Integer
Declare Function Istrcmpi Lib "User" (ByVal lpString1 As Any, ByVal lpString2 As Any) As Integer
Declare Function Istrcpy Lib "Kernel" (ByVal lpString1 As Any, ByVal lpString2 As Any) As Long
Declare Function Istrcat Lib "Kernel" (ByVal lpString1 As Any, ByVal lpString2 As Any) As Long
Declare Function Istrlen Lib "Kernel" (ByVal lpString As Any) As Integer
Declare Function ToAscii Lib "Keyboard" (ByVal wVirtKey As Integer, ByVal wScanCode As Integer, lpKeyState As Any, lpChar As Any, ByVal wFlags As Integer) As Integer
Declare Function wvsprintf Lib "User" (ByVal lpOutput As String, ByVal lpFormat As String, lpArglist As Integer) As Integer
' vsprintf can not be called from Visual Basic
' Declare Function wsprintf Lib "User" (ByVal lpOutput As String, ByVal lpFormat As String) As Integer
```

'Atom Management Functions

```
Declare Function InitAtomTable Lib "Kernel" (ByVal nSize As Integer) As Integer
Declare Function AddAtom Lib "Kernel" (ByVal lpString As String) As Integer
Declare Function DeleteAtom Lib "Kernel" (ByVal nAtom As Integer) As Integer
Declare Function FindAtom Lib "Kernel" (ByVal lpString As String) As Integer
Declare Function GetAtomName Lib "Kernel" (ByVal nAtom As Integer, ByVal lpBuffer As String, ByVal nSize As Integer) As Integer
Declare Function GlobalAddAtom Lib "Kernel" (ByVal lpString As String) As Integer
Declare Function GlobalDeleteAtom Lib "Kernel" (ByVal nAtom As Integer) As Integer
Declare Function GlobalFindAtom Lib "Kernel" (ByVal lpString As String) As Integer
Declare Function GlobalGetAtomName Lib "Kernel" (ByVal nAtom As Integer, ByVal lpbuffer As String, ByVal nSize As Integer) As Integer
Declare Function GetAtomHandle Lib "Kernel" (ByVal wAtom As Integer) As Integer
Declare Function MakeIntAtom Lib "Kernel" (ByVal wInteger As Long) As String
```

'Initialization File Functions

Declare Function GetProfileInt Lib "Kernel" (ByVal IpAppName As String, ByVal IpKeyName As String, ByVal nDefault As Integer) As Integer

Declare Function GetProfileString Lib "Kernel" (ByVal IpAppName As String, ByVal IpKeyName As String, ByVal IpDefault As String, ByVal IpReturnedString As String, ByVal nSize As Integer) As Integer

Declare Function WriteProfileString Lib "Kernel" (ByVal IpApplicationName As String, ByVal IpKeyName As String, ByVal IpString As String) As Integer

Declare Function GetPrivateProfileInt Lib "Kernel" (ByVal IpApplicationName As String, ByVal IpKeyName As String, ByVal nDefault As Integer, ByVal IpFileName As String) As Integer

Declare Function GetPrivateProfileString Lib "Kernel" (ByVal IpApplicationName As String, ByVal IpKeyName As String, ByVal IpDefault As String, ByVal IpReturnedString As String, ByVal nSize As Integer, ByVal IpFileName As String) As Integer

Declare Function WritePrivateProfileString Lib "Kernel" (ByVal IpApplicationName As String, ByVal IpKeyName As String, ByVal IpString As String, ByVal IpFileName As String) As Integer

'Communications Functions

```
Declare Function OpenComm Lib "User" (ByVal lpComName As String, ByVal wInQueue As Integer,
ByVal wOutQueue As Integer) As Integer
Declare Function SetCommState Lib "User" (lpDCB As DCB) As Integer
Declare Function GetCommState Lib "User" (ByVal nCid As Integer, lpDCB as DCB) As Integer
Declare Function ReadComm Lib "User" (ByVal nCid As Integer, ByVal lpBuf As String, ByVal nSize As
Integer) As Integer
Declare Function UngetCommChar Lib "User" (ByVal nCid As Integer, ByVal cChar As Integer) As Integer
Declare Function WriteComm Lib "User" (ByVal nCid As Integer, ByVal lpBuf As String, ByVal nSize As
Integer) As Integer
Declare Function CloseComm Lib "User" (ByVal nCid As Integer) As Integer
Declare Function BuildCommDCB Lib "User" (ByVal lpDef As String, lpDCB As DCB) As Integer
Declare Function TransmitCommChar Lib "User" (ByVal nCid As Integer, ByVal cChar As Integer) As
Integer
Declare Function SetCommEventMask Lib "User" (ByVal nCid as Integer, nEvtMask as Integer) As Long
Declare Function GetCommEventMask Lib "User" (ByVal nCid As Integer, ByVal nEvtMask As Integer) As
Integer
Declare Function SetCommBreak Lib "User" (ByVal nCid As Integer) As Integer
Declare Function ClearCommBreak Lib "User" (ByVal nCid As Integer) As Integer
Declare Function FlushComm Lib "User" (ByVal nCid As Integer, ByVal nQueue As Integer) As Integer
Declare Function EscapeCommFunction Lib "User" (ByVal nCid As Integer, ByVal nFunc As Integer) As
Integer
Declare Function GetCommError Lib "User" (ByVal nCid As Integer, lpStat As COMSTAT) As Integer
```

```
' COMM declarations
```

```
,
```

```
Type DCB
```

```
    Id As String * 1
    BaudRate As Integer
    ByteSize As String * 1
    Parity As String * 1
    StopBits As String * 1
    RlsTimeout As Integer
    CtsTimeout As Integer
    DsrTimeout As Integer
    Bits1 As String * 1 ' The fifteen actual DCB bit-sized data fields
    Bits2 As String * 1 ' within these two bytes can be manipulated by
                        ' bitwise logical And/Or operations. Refer to
                        ' SDKWIN.HLP for location/meaning of specific bits
```

```
    XonChar As String * 1
    XoffChar As String * 1
    XonLim As Integer
    XoffLim As Integer
    PeChar As String * 1
    EofChar As String * 1
    EvtChar As String * 1
    TxDelay As Integer
```

```
End Type
```

```
Type COMSTAT
```

```
    Bits As String * 1 ' For specific bit flags and their
                        ' meanings, refer to SDKWIN.HLP.
    cbInQue As Integer
    cbOutQue As Integer
```

End Type

Global Const NOPARITY = 0
Global Const ODDPARITY = 1
Global Const EVENPARITY = 2
Global Const MARKPARITY = 3
Global Const SPACEPARITY = 4

Global Const ONESTOPBIT = 0
Global Const ONE5STOPBITS = 1
Global Const TWOSTOPBITS = 2

Global Const IGNORE = 0 ' Ignore signal
Global Const INFINITE = &HFFFF ' Infinite timeout

' Error Flags

Global Const CE_RXOVER = &H1 ' Receive Queue overflow
Global Const CE_OVERRUN = &H2 ' Receive Overrun Error
Global Const CE_RXPARITY = &H4 ' Receive Parity Error
Global Const CE_FRAME = &H8 ' Receive Framing error
Global Const CE_BREAK = &H10 ' Break Detected
Global Const CE_CTSTO = &H20 ' CTS Timeout
Global Const CE_DSRTO = &H40 ' DSR Timeout
Global Const CE_RLSDTO = &H80 ' RLSD Timeout
Global Const CE_TXFULL = &H100 ' TX Queue is full
Global Const CE_PTO = &H200 ' LPTx Timeout
Global Const CE_IOE = &H400 ' LPTx I/O Error
Global Const CE_DNS = &H800 ' LPTx Device not selected
Global Const CE_OOP = &H1000 ' LPTx Out-Of-Paper
Global Const CE_MODE = &H8000 ' Requested mode unsupported

Global Const IE_BADID = (-1) ' Invalid or unsupported id
Global Const IE_OPEN = (-2) ' Device Already Open
Global Const IE_NOOPEN = (-3) ' Device Not Open
Global Const IE_MEMORY = (-4) ' Unable to allocate queues
Global Const IE_DEFAULT = (-5) ' Error in default parameters
Global Const IE_HARDWARE = (-10) ' Hardware Not Present
Global Const IE_BYTESIZE = (-11) ' Illegal Byte Size
Global Const IE_BAUDRATE = (-12) ' Unsupported BaudRate

' Events

Global Const EV_RXCHAR = &H1 ' Any Character received
Global Const EV_RXFLAG = &H2 ' Received certain character
Global Const EV_TXEMPTY = &H4 ' Transmitt Queue Empty
Global Const EV_CTS = &H8 ' CTS changed state
Global Const EV_DSR = &H10 ' DSR changed state
Global Const EV_RLSD = &H20 ' RLSD changed state
Global Const EV_BREAK = &H40 ' BREAK received
Global Const EV_ERR = &H80 ' Line status error occurred
Global Const EV_RING = &H100 ' Ring signal detected
Global Const EV_PERR = &H200 ' Printer error occurred

' Escape Functions

Global Const SETXOFF = 1 ' Simulate XOFF received
Global Const SETXON = 2 ' Simulate XON received
Global Const SETRTS = 3 ' Set RTS high

Global Const CLRRTS = 4 ' Set RTS low
Global Const SETDTR = 5 ' Set DTR high
Global Const CLRDTR = 6 ' Set DTR low
Global Const RESETDEV = 7 ' Reset device if possible

Global Const LPTx = &H80 ' Set if ID is for LPT device

'Sound Functions

```
Declare Function OpenSound Lib "Sound" () As Integer
Declare Sub CloseSound Lib "Sound" ()
Declare Function SetVoiceQueueSize Lib "Sound" (ByVal nVoice As Integer, ByVal nBytes As Integer) As Integer
Declare Function SetVoiceNote Lib "Sound" (ByVal nVoice As Integer, ByVal nValue As Integer, ByVal nLength As Integer, ByVal nCdots As Integer) As Integer
Declare Function SetVoiceAccent Lib "Sound" (ByVal nVoice As Integer, ByVal nTempo As Integer, ByVal nVolume As Integer, ByVal nMode As Integer, ByVal nPitch As Integer) As Integer
Declare Function SetVoiceEnvelope Lib "Sound" (ByVal nVoice As Integer, ByVal nShape As Integer, ByVal nRepeat As Integer) As Integer
Declare Function SetSoundNoise Lib "Sound" (ByVal nSource As Integer, ByVal nDuration As Integer) As Integer
Declare Function SetVoiceSound Lib "Sound" (ByVal nVoice As Integer, ByVal nFrequency As Long, ByVal nDuration As Integer) As Integer
Declare Function StartSound Lib "Sound" () As Integer
Declare Function StopSound Lib "Sound" () As Integer
Declare Function WaitSoundState Lib "Sound" (ByVal nState As Integer) As Integer
Declare Function SyncAllVoices Lib "Sound" () As Integer
Declare Function CountVoiceNotes Lib "Sound" (ByVal nVoice As Integer) As Integer
Declare Function GetThresholdEvent Lib "Sound" () As Integer
Declare Function GetThresholdStatus Lib "Sound" () As Integer
Declare Function SetVoiceThreshold Lib "Sound" (ByVal nVoice As Integer, ByVal nNotes As Integer) As Integer
```

' WaitSoundState() Constants

```
Global Const S_QUEUEEMPTY = 0
Global Const S_THRESHOLD = 1
Global Const S_ALLTHRESHOLD = 2
```

' Accent Modes

```
Global Const S_NORMAL = 0
Global Const S_LEGATO = 1
Global Const S_STACCATO = 2
```

' SetSoundNoise() Sources

```
Global Const S_PERIOD512 = 0      ' Freq = N/512 high pitch, less coarse hiss
Global Const S_PERIOD1024 = 1    ' Freq = N/1024
Global Const S_PERIOD2048 = 2    ' Freq = N/2048 low pitch, more coarse hiss
Global Const S_PERIODVOICE = 3   ' Source is frequency from voice channel (3)
Global Const S_WHITE512 = 4      ' Freq = N/512 high pitch, less coarse hiss
Global Const S_WHITE1024 = 5     ' Freq = N/1024
Global Const S_WHITE2048 = 6     ' Freq = N/2048 low pitch, more coarse hiss
Global Const S_WHITEVOICE = 7    ' Source is frequency from voice channel (3)
```

```
Global Const S_SERDVNA = (-1)    ' Device not available
Global Const S_SEROFM = (-2)    ' Out of memory
Global Const S_SERMACT = (-3)   ' Music active
Global Const S_SERQFUL = (-4)   ' Queue full
Global Const S_SERBDNT = (-5)   ' Invalid note
Global Const S_SERDLN = (-6)    ' Invalid note length
Global Const S_SERDCC = (-7)    ' Invalid note count
Global Const S_SERDTP = (-8)    ' Invalid tempo
Global Const S_SERDVL = (-9)    ' Invalid volume
Global Const S_SERDMD = (-10)   ' Invalid mode
```

Global Const S_SERDSH = (-11)	' Invalid shape
Global Const S_SERDPT = (-12)	' Invalid pitch
Global Const S_SERDFQ = (-13)	' Invalid frequency
Global Const S_SERDDR = (-14)	' Invalid duration
Global Const S_SERDSR = (-15)	' Invalid source
Global Const S_SERDST = (-16)	' Invalid state

'Utility Functions and Macros

Declare Function MulDiv Lib "GDI" (ByVal nNumber As Integer, ByVal nNumerator As Integer, ByVal nDenominator As Integer) As Integer

' HIBYTE, HIWORD, LOBYTE, LOWORD, MAKEINTATOM, MAKEINTRESOURCE, MAKELONG, MAKEPOINT, PALETTEINDEX, PALETTERGB and RGB are 'macros and can not be used within Visual Basic.

'File I/O Functions

Declare Function GetDriveType Lib "Kernel" (ByVal nDrive As Integer) As Integer

' GetDriveType return values
Global Const DRIVE_REMOVABLE = 2
Global Const DRIVE_FIXED = 3
Global Const DRIVE_REMOTE = 4

Declare Function GetWindowsDirectory Lib "Kernel" (ByVal lpBuffer As String, ByVal nSize As Integer) As Integer

Declare Function GetSystemDirectory Lib "Kernel" (ByVal lpBuffer As String, ByVal nSize As Integer) As Integer

Declare Function OpenFile Lib "Kernel" (ByVal lpFileName As String, lpReOpenBuff As OFSTRUCT, ByVal wStyle As Integer) As Integer

' OpenFile() Structure
Type OFSTRUCT
 cBytes As String * 1
 fFixedDisk As String * 1
 nErrCode As Integer
 r1 As Integer 'file date
 r2 As Integer 'file time
 szPathName As String * 128
End Type

' OpenFile() Flags
Global Const OF_READ = &H0
Global Const OF_WRITE = &H1
Global Const OF_READWRITE = &H2
Global Const OF_SHARE_COMPAT = &H0
Global Const OF_SHARE_EXCLUSIVE = &H10
Global Const OF_SHARE_DENY_WRITE = &H20
Global Const OF_SHARE_DENY_READ = &H30
Global Const OF_SHARE_DENY_NONE = &H40
Global Const OF_PARSE = &H100
Global Const OF_DELETE = &H200
Global Const OF_VERIFY = &H400
Global Const OF_CANCEL = &H800
Global Const OF_CREATE = &H1000
Global Const OF_PROMPT = &H2000
Global Const OF_EXIST = &H4000
Global Const OF_REOPEN = &H8000

Declare Function GetTempDrive Lib "Kernel" (ByVal cDriveLetter as Integer) As Integer

' GetTempFileName() Flags

Global Const TF_FORCEDRIVE = &H80

Declare Function GetTempFileName Lib "Kernel" (ByVal cDriveLetter as Integer, ByVal lpPrefixString As String, ByVal wUnique As Integer, ByVal lpTempFileName As String) As Integer

Declare Function SetHandleCount Lib "Kernel" (ByVal wNumber As Integer) As Integer

Declare Function lopen Lib "Kernel" Alias "_lopen" (ByVal lpPathName As String, ByVal iReadWrite As Integer) As Integer

```
Global Const READAPI = 0      ' Flags for _lopen
Global Const WRITEAPI = 1
Global Const READ_WRITE = 2
```

```
Declare Function lclose Lib "Kernel" Alias "_lclose" (ByVal hFile As Integer) As Integer
Declare Function lcreat Lib "Kernel" Alias "_lcreat" (ByVal lpPathName As String, ByVal iAttribute As Integer) As Integer
Declare Function llseek Lib "Kernel" Alias "_llseek" (ByVal hFile As Integer, ByVal lOffset As Long, ByVal iOrigin As Integer) As Long
Declare Function lread Lib "Kernel" Alias "_lread" (ByVal hFile As Integer, ByVal lpBuffer As String, ByVal wBytes As Integer) As Integer
Declare Function lwrite Lib "Kernel" Alias "_lwrite" (ByVal hFile As Integer, ByVal lpBuffer As String, ByVal wBytes As Integer) As Integer
```

'Debugging Functions

```
Declare Sub DebugBreak Lib "Kernel" ()  
Declare Sub OutputDebugString Lib "Kernel" (ByVal lpOutputString As String)  
Declare Sub FatalExit Lib "Kernel" (ByVal Code As Integer)  
Declare Function ValidateFreeSpaces Lib "Kernel" () As Long  
Declare Sub ValidateCodeSegments Lib "Kernel" ()  
Declare Sub FatalAppExit Lib "Kernel" (ByVal wAction As Long, ByVal lpMessageText As String)
```

'Optimization Tool Functions

```
Declare Function ProfInsChk Lib "User" () As Integer
Declare Sub ProfSetup Lib "User" (ByVal nBufferSize As Integer, ByVal nSamples As Integer)
Declare Sub ProfSampRate Lib "User" (ByVal nRate286 As Integer, ByVal nRate386 As Integer)
Declare Sub ProfStart Lib "User" ()
Declare Sub ProfStop Lib "User" ()
Declare Sub ProfClear Lib "User" ()
Declare Sub ProfFlush Lib "User" ()
Declare Sub ProfFinish Lib "User" ()
Declare Sub SwapRecording Lib "Kernel" (ByVal wFlag As Integer)
```

'Application Execution Functions

Declare Function WinExec Lib "Kernel" (ByVal lpCmdLine As String, ByVal nCmdShow As Integer) As Integer

Declare Function LoadModule Lib "Kernel" (ByVal lpModuleName As String, lpParameterBlock As PARAMETERBLOCK) As Integer

Declare Function WinHelp Lib "User" (ByVal hWnd As Integer, ByVal lpHelpFile As String, ByVal wCommand As Integer, dwData As Any) As Integer

' Commands to pass WinHelp()

Global Const HELP_CONTEXT = &H1 ' Display topic in ulTopic

Global Const HELP_QUIT = &H2 ' Terminate help

Global Const HELP_INDEX = &H3 ' Display index

Global Const HELP_HELPONHELP = &H4 ' Display help on using help

Global Const HELP_SETINDEX = &H5 ' Set the current Index for multi index help

Global Const HELP_KEY = &H101 ' Display topic for keyword in offabData

Global Const HELP_MULTIKEY = &H201

Type MULTIKEYHELP

mkSize As Integer

mkKeylist As String * 1

szKeyphrase As String * 253 ' Array length is arbitrary; may be changed

End Type