

ShapedDemo2.bas

This code accompanies the article [Creating a Nonrectangular Window](#)

```
'ShapedDemo2.bas - Janet Terra
'Demo to accompany
'Demo - Creating a Nonrectangular Window
'LBPE July, 2011
'Originally appeared in
'LB Newsletter #132, May, 2005

    Nomainwin

'Define the Window
    WindowWidth = 250
    WindowHeight = 250
    UpperLeftX = int((DisplayWidth-WindowWidth)/2)
    UpperLeftY = int((DisplayHeight-WindowHeight)/2)

    stylebits #ShapeWindow.gb, 0, _WS_BORDER, 0, 0
    graphicbox #ShapeWindow.gb, 0, 0, 250, 250
    stylebits #ShapeWindow, 0, 0, _WS_EX_TOPMOST, 0

    open "Shape Window" for window_popup as #ShapeWindow
    #ShapeWindow "trapclose [closeShapeWindow]"

'Obtain the Handles and Device Controls
    hBw = hWnd(#ShapeWindow)
    hBgb = hWnd(#ShapeWindow.gb)
    hDCw = GetDC(hBw)
    hDCgb = GetDC(hBgb)

'Draw the Shape
    #ShapeWindow.gb "down; fill black"
    gosub [drawShape]
    wait

[closeShapeWindow]
    call DelObject hBw
    close #ShapeWindow
end
```

```
[drawShape]
'Set region to null
    hRgn = RectRegion(0, 0, 0, 0)

'Draw a Circle
    #ShapeWindow.gb "color darkred; backcolor red"
    #ShapeWindow.gb "place 100 100; circlefilled 75"

'Draw a Rectangle
    #ShapeWindow.gb "color darkblue; backcolor blue"
    #ShapeWindow.gb "place 50 200; boxfilled 225 225"

'Set background to Transparent
    call SetBkMode hDCgb, 1
'Release memory
    call ReleaseDC hBgb, hDCbg

'Format and write text
    #ShapeWindow.gb "font Courier_New 16 86 Bold"
    #ShapeWindow.gb "color Darkgreen; place 5 210"
    #ShapeWindow.gb "\Alt-F4 to Close"
    #ShapeWindow.gb "flush"

'Read each pixel.  Add each pixel to hRgn only if
'color is NOT black (0)
    for x = 0 to 250
        for y = 0 to 250
            if pixelColor(hDCgb, x, y) <> 0 then
                hTempRgn = RectRegion(x, y, x+1, y+1)
                newRgn = CombineRgn(hRgn, hRgn, hTempRgn, 3)
                call DelObject hTempRgn
            end if
        next y
    next x

'Set the region as the Window
    call SetWindowRgn hBw, hRgn, 1
    return

function GetDC(hW)
    call dll #user32, "GetDC", _
        hW as ulong, _
        GetDC as ulong
end function

function RectRegion(ulx, uly, width, height)
```

```
    calldll #gdi32, "CreateRectRgn", _
        ulx as long, _
        uly as long, _
        width as long, _
        height as long, _
        RectRegion as ulong
end function

function CombineRgn(hDest, hSource1, hSource2, combineMode)
    calldll #gdi32, "CombineRgn", _
        hDest as ulong, _
        hSource1 as ulong, _
        hSource2 as ulong, _
        combineMode as long, _
        CombineRgn as ulong
end function

sub DelObject hObject
    calldll #gdi32, "DeleteObject", _
        hObject as ulong, _
        null as long
end sub

sub ReleaseDC hWnd, hDC
    calldll #user32, "ReleaseDC", _
        hWnd as ulong, _
        hDC as ulong, _
        null as long
end sub

sub SetWindowRgn hWnd, hRgn, redrawMode
    calldll #user32, "SetWindowRgn", _
        hWnd as ulong, _
        hRgn as ulong, _
        redrawMode as long, _
        SetWindowRgn as long
end sub

sub SetBkMode hDC, flag
    calldll #gdi32, "SetBkMode", _
        hDC as ulong, _
        flag as long, _
        null as long
end sub

function pixelColor(hDC, x, y)
```

```
    calldll #gdi32, "GetPixel", _  
        hDC as ulong, _  
        x as long, _  
        y as long, _  
        pixelColor as long  
end function
```