

Forum:[Hack Forums](#)

Thread: [WndProc for custom button\[Win32api\]](#)

aeiro - July 17, 2018, 6:13 pm

This is my first time creating and handling a custom button from registering a new window class to drawing the button in the new wndproc function.

Therefore i felt it'd be nice to share. The for loops need a bit of work to fully fill the center of the button and fonts act odd with certain colors. I havn't really figured that part.

I'm also still building this proc to hopefully convert it into an edit control.

Currently these buttons can only do what you see in the below GIF change color on hover and stay colored on click with unclick other active button.

PHP Code

```
<?php
LRESULT CALLBACK cb::WndProc2(HWND hWnd, UINT message, WPARAM wParam,
LPARAM lParam)
{
    static BOOL BMT = FALSE;
    static BOOL LBD = FALSE;
    int menu = (int)GetMenu(hWnd);
    switch (message)
    {

    case WM_MOUSEMOVE:
    {
        if (!BMT)
        {
            TRACKMOUSEEVENT TME = { NULL };
            TME.cbSize = sizeof(TRACKMOUSEEVENT);
            TME.dwFlags = TME_LEAVE;
            TME.hwndTrack = hWnd;
            TME.dwHoverTime = NULL;
            BMT = TrackMouseEvent(&TME);
            InvalidateRect(hWnd, 0, TRUE);
        }

        break;
    }
    case WM_MOUSELEAVE:
    {

        BMT = FALSE;
        InvalidateRect(hWnd, 0, TRUE);
        break;
    }
    case WM_LBUTTONDOWN:
    {

        if (!LBD)
```

```

LBD = menu;
else if (LBD == menu)
{
LBD = FALSE;
InvalidateRect(hWnd, 0, TRUE);
}
else
{
int button = LBD;
HWND parent = GetParent(hWnd);
HWND curbutton = GetDlgItem(parent, button);
LBD = menu;
BMT = FALSE;
InvalidateRect(curbutton, 0, TRUE);

```

```

break;
}
case WM_CREATE:
{
RECT r = cb::GetRect(hWnd);
int thirdWidth = r.right / 5;
int thirdHeight = r.bottom / 5;

```

```

break;
}

```

```

break;
case WM_PAINT:
{
PAINTSTRUCT ps;
HDC hdc = BeginPaint(hWnd, &ps);
RECT r = cb::GetRect(hWnd);
HPEN hp;

```

```

int menu = (int)GetMenu(hWnd);
hp = CreatePen(PS_SOLID, 4, BMT || (LBD == menu ? 1 : 0) ? RGB(0x00, 0xb1, 0x21) : RGB(200, 200, 200));
SelectObject(hdc, hp);

```

```

int radius = r.bottom / 2;

```

```

MoveToEx(hdc, r.left + radius, 0, 0);
LineTo(hdc, r.right - radius, 0);
MoveToEx(hdc, r.left + radius, r.bottom, 0);
LineTo(hdc, r.right - radius, r.bottom);

```

```

int x, y, r2 = 0;
r2 = radius * radius;
for (x = 0; x <= radius; x++)

```

```

{
y = (int)(sqrt((r2 - x * x) + .5));

MoveToEx(hdc, radius + -x, radius + -y, 0);
LineTo(hdc, r.right - radius + x, radius + -y);
MoveToEx(hdc, radius + -x, radius + y, 0);
LineTo(hdc, r.right - radius + x, radius + y);
MoveToEx(hdc, r.right - radius + x, radius + -y, 0);
LineTo(hdc, radius + -x, radius + -y);
MoveToEx(hdc, r.right - radius + x, radius + y, 0);
LineTo(hdc, radius + -x, radius + y);

}

int fillend = radius + (r.bottom / 12);
int fillup = radius - (r.bottom / 12);
MoveToEx(hdc, r.left, fillend, 0);
LineTo(hdc, r.right, fillup);

MoveToEx(hdc, r.right, fillend, 0);

LineTo(hdc, r.left, fillup);

SetTextColor(hdc, RGB(0, 45, 25));

WCHAR staticText[99];
int len = SendMessage(hWnd, WM_GETTEXT,
ARRAYSIZE(staticText), (LPARAM)staticText);
int thirdWidth = r.right / 3;
int thirdHeight = r.bottom / 4;

HANDLE hFontTa = CreateFont(20, 20, 0, 0, FW_EXTRABOLD, FALSE, FALSE, 0,
ANSI_CHARSET, OUT_DEFAULT_PRECIS, CLIP_DEFAULT_PRECIS,
PROOF_QUALITY, DEFAULT_PITCH | FF_DONTCARE, L"Adobe Devanagari Regular");
SelectObject(hdc, hFontTa);
SetBkColor(hdc, BMT || (LBD == menu ? 1 : 0) ? RGB(0x00, 0xb1, 0x21) : RGB(200, 200, 200));
TextOut(hdc, thirdWidth, thirdHeight, staticText, len);

ReleaseDC(hWnd, hdc);

EndPaint(hWnd, &ps);
}
break;
case WM_DESTROY:
PostQuitMessage(0);
break;
default:
return DefWindowProc(hWnd, message, wParam, lParam);
}
return 0;
}

```

maryous - July 18, 2018, 9:09 am

what is the use of "MoveToEx and LineTo" if you don't mind me asking

aeiro - July 18, 2018, 1:48 pm

(Jul 18, 2018 09:09 AM)maryous Wrote: [\[CITE\]](#) what is the use of "MoveToEx and LineTo" if you don't mind me asking

draws a line from the points specified. you first set the starting point of the line with moveto then the endpoint with lineto which completes the draw.

maryous - July 18, 2018, 2:30 pm

(Jul 18, 2018 01:48 PM)aeiro Wrote: [\[CITE\]](#) draws a line from the points specified. you first set the starting point of the line with moveto then the endpoint with lineto which completes the draw.

thanks ? ? ?

aeiro - July 22, 2018, 10:43 am