

These functions will (hopefully) make progress bars easier.

```
'PROGRESS BAR CONSTANTS, MUST BE DEFINED!
Global PBM.SETSTEP : PBM.SETSTEP = 1028
Global PBM.STEPIT : PBM.STEPIT = 1029
Global PBM.SETPOS : PBM.SETPOS = (_WM_USER) + 2
'END CONSTANTS

WindowWidth = 150
WindowHeight = 120

nomainwin
button #1.b, "Step It!", [step], UL, 20, 50, 50, 25
open "window" for window as #1

#1, "trapclose [quit]"
hWnd = hWnd(#1)
hProg = CreateProgressBar(hWnd, 20, 20, 100, 25)
Call SetStep hProg, 10
wait

[step]
If x >= 10 then notice "Progress Bar Full!" : wait
Call StepIt hProg
x = x + 1
wait

[quit]
call DestroyProgressBar hProg
close #1
end

'PROGRESS BAR FUNCTIONS BELOW!
Function CreateProgressBar(hWnd, x, y, w, h)
    extStyle = _WS_EX_CLIENTEDGE
    progStyle = _WS_CHILD or _WS_VISIBLE

    CallDLL #user32, "GetWindowLongA",_
    hWnd as ulong,_
    _GWL_HINSTANCE as long,_
    hInst as ulong

    CallDLL #user32, "CreateWindowExA",_
```

```
extStyle as long,_
"msctls_progress32" as ptr,_
" " as ptr,_
progStyle as long,_
x as long,_
y as long,_
w as long,_
h as long,_
hWnd as ulong,_
0 as long,_
hInst as ulong,_
CreateProgressBar as ulong
End Function

Sub SetStep hProg, percent
CallDLL #user32, "SendMessageA",_
hProg as ulong,_
PBM.SETSTEP as long,_
percent as long,_
0 as long,_
ret as long
End Sub

Sub StepIt hProg
CallDLL #user32, "SendMessageA",_
hProg as ulong,_
PBM.STEPIT as long,_
0 as long,_
0 as long,_
ret as long
End Sub

Sub SetPos hProg, num
CallDLL #user32, "SendMessageA",_
hProg as ulong,_
PBM.SETPOS as long,_
num as long,_
0 as long,_
ret as long
End Sub

Sub DestroyProgressBar hProg
CallDLL #user32, "DestroyWindow",_
hProg as ulong,_
ret as boolean
End Sub
```

They should be easy to use. Syntaxes:

NOTE: In ALL of the syntaxes, hWnd should be the window handle, and hProg should be the handle of the Progress bar, returned from the CreateProgressBar function.

```
hProg = CreateProgressBar(hWnd, X, Y, W, H)
```

Where X, Y, W, and H are the position the progress bar should be.

SetStep:

```
Call SetStep hProg, PERCENT
```

Replace PERCENT with the number that you would like the progress bar to increment each time you tell it to, in percentage.

StepIt:

```
Call StepIt hProg
```

This just increments the progress bar by whatever you sent to the SetStep sub.

SetPos:

```
Call SetPos hProg, NUM
```

Where NUM is the number you want the progress bar to jump to.

[thedarkfreak](#) Mar 11, 2008 (Chris Iverson)

Should be easy enough ;)