Gray Scale Palette Example

```
//  Gray Scale Palette Example
//  Since this is example code, it contains none of the standard error checks and
//  user interface expected of a real program.  It is only intended as a starting
//  point.

//  This code draws a grayscale bullseye using the palette manager.  It will work
//  with 4 or 8 bit screens.  Although it modifies the gDevice's color table, it
//  will restore it before exiting.  This code will *not* work unless you have
//  32 bit QuickDraw.


// Assumes inclusion of <MacHeaders>
#include <Palettes.h>

void draw (void);
void init (void);
void SetGrayPalette (WindowPtr w);
void checkWorld (void);
void die (char *s);

#define ScreenDepth(gdh)((*((*gdh)->gdPMap))->pixelSize)
#define MaxColors(gdh)          (1<<ScreenDepth(gdh))

WindowPtr mainWindow;
GDHandle        gMyGDevice;

main()
{
    Rect    windRect;

    checkWorld();
    init();

    gMyGDevice = GetGDevice();

    windRect = thePort->portBits.bounds;
    InsetRect(&windRect, 50, 50);
    mainWindow = (WindowPtr)NewCWindow(nil, &windRect, "\pUntitled",
                                       TRUE, documentProc,
                                       (WindowPtr)-1, FALSE, 0);
    SetPort(mainWindow);                    // set window to current graf port
    SetGrayPalette(mainWindow);
    InitMenus();
    draw();
    while (!Button())
        ;
    RestoreDeviceClut(gMyGDevice);
}

void draw()
{
    Rect    tempRect;
    short   i, colors = MaxColors(gMyGDevice);
```

```
        SetRect(&tempRect, colors, colors, colors, colors);
        for (i = 0; i < colors - 2; i++) {
                PmForeColor(i);
                FrameOval (&tempRect);
                InsetRect (&tempRect, -1, -1);
        }
}

void init()
{
        InitGraf(&thePort);
        InitFonts();
        InitWindows();
        TEInit();
        InitDialogs(nil);
        InitCursor();
}

#define EXACT       0                                   // used with pmTolerant,
// only exact matches

#define GRAY_RAMP       32              // 32bit quickdraw has a built-in gray
// ramp, located
                                        // at (32 + <the bit depth of the ramp>)

void SetGrayPalette(WindowPtr w)
{
        CTabHandle      tab = GetCTable(GRAY_RAMP + ScreenDepth(gMyGDevice));
        PaletteHandle   newPal = NewPalette((*tab)->ctSize, tab, pmTolerant,
                                        EXACT);

        NSetPalette(w, newPal, pmAllUpdates);
        ActivatePalette(w);
}

#define     QD32Trap    0xAB03
#define     UnImplTrap  0xA89F

void checkWorld()
{
        OSErr           err;
        SysEnvRec       world;

        err = SysEnvirons(2, &world);
        if (!world.hasColorQD)
                die("no color qd");
        if (NGetTrapAddress(QD32Trap, ToolTrap) ==
                        NGetTrapAddress(UnImplTrap,ToolTrap))
                die("no 32bit qd");
}

void die(char *s)
{
        CtoPstr(s);
        DebugStr(s);
```

```
    ExitToShell();
}
```