

```

//*****
//*  MACRO: Palette.wcm
//*  PURPOSE: To create a printable color palette.
//*****
Application (A1; "WPPrWin"; Default; "US")
PRActivate()

Index := 0
Width := 700
Height := 450
Margin := 600
vPaletteName := ""

// Get users choice of palette; Default, Current, Select
NewDrawingCreate()
VarErrChk(On!)
DialogDefine("PaletteSelection"; 50; 50; 155; 110; Percent!; "Palette Selection")
    DialogAddGroupBox("PaletteSelection"; -1; 10; 10; 75; 75; "Choose Palette")
    DialogAddRadioButton("PaletteSelection"; 100; 20; 25; 55; 15; "Default"; vRadio1)
    DialogAddRadioButton("PaletteSelection"; 101; 20; 45; 55; 15; "Current"; vRadio2)
    DialogAddRadioButton("PaletteSelection"; 102; 20; 65; 55; 15; "Select Other"; vRadio3)
    DialogAddPushButton("PaletteSelection"; 103; 105; 15; 35; 13; DefaultBtn!; "OK")
    DialogAddPushButton("PaletteSelection"; 104; 105; 35; 35; 13; NonDefaultBtn!; "Cancel")
DialogDisplay("PaletteSelection"; 100)
If(MacroDialogResult <> 103)                // If not OK, quit.
    DialogDestroy("PaletteSelection")
    Quit
Endif
If(vRadio1 <> 0)                            // Default Palette
    Selection := 1
Else If(vRadio2 <> 0)
    Selection := 2                        // Current Palette
Else If(vRadio3 <> 0)
    Selection := 3                        // Select a Palette
Endif
Endif
DialogDestroy("PaletteSelection")

// Set current or palette name(select)
Switch (Selection)
    CaseOf 1:                            // Set the palette to Default
        ResetColorTable()
    CaseOf 3:                            // Select a palette
        Call(SelectPalette)
EndSwitch

//Create Palette
Call(CreatePalette)
Call(AddText)
MessageBox(x; "Printing Note"; "The state of Adjust Image to Print Black and White in the " +
    "Output Setting Section of the Print Dialog will have a dramatic effect on the colours " +
    "printed. You will need to change this setting to see which method of printing works " +
    "best for you."; IconExclamation!)
Quit

Label(CreatePalette)
    BeginSelect()
    LineAttributes (Yes!; Both!; Yes!; No!; Yes!; Alternating!; Flat!; Flat!; Bevel!; Solid!; Solid!; 0; 0; 0; 0; 0; 0; 0)
    ObjectSetLineWidth (LineWidth: 0w)
    for (y; 0; y < 16; y + 1)

```

```

    for (x; 0; x < 16; x + 1)
        Index := y * 16 + x
        yLoc = (15 * Height) - (y * Height) + Margin + (Height/2)
        xLoc = (x * Width + Margin)
        ObjectSetFillForeground (EnvColorPalette (Red!; Index);
                                EnvColorPalette (Green!; Index);
                                EnvColorPalette (Blue!; Index);
                                Transparent: 0)
        AddPolyLine (Both!; 4; {xLoc; yLoc; xLoc + Width; yLoc;
                                xLoc + Width; yLoc + Height; xLoc; yLoc + Height})
    endfor
endfor
SelectAllObjects ()
GroupObjects ()
AlignObjectsRight ()
UnselectAllObjects()
Return

Label(AddText)
// Reset text settings to default
SetFontFace ("Times New Roman Regular"; 36; Off!; Off!; Off!; Normal!; Normal!; TT!)
SetFontPattern (Solid!; 0; 0; 0; 255; 255; 255; 0; 0; 0; 0; 0; 0; 0; 0; 0; FramedFilled!)
SetFontGradient (None!; 64; 0; 0; 0; 1; 0; 0; 0; 255; 255; 255; 255; 255; 255; 0; 0)
RealizeFontChanges (Both!)

// Begin adding the text
TextLineCreate ()
ObjectSetFillForeground (0; 0; 0; 0)
AddTextBox (LeftEdge: 1380w; BottomEdge: 8900w; RightEdge: 1380w; TopEdge: 8900w)
Switch(Selection)
    CaseOf 1:      Type("Default")
    CaseOf 2:      Type("Current")
    CaseOf 3:      Type(vPaletteName)
EndSwitch
Type (Text: " Colour Palette")
HardReturn ()
SetFontFace(PointSize:24)
RealizeFontChanges()
yLoc = (17 * Height) + Margin + (Height/2)
for (x; 0; x < 16; x + 1)
    xLoc = (x * Width + 2.5*Margin)
    TextBoxCreate ()
    AddTextBox
        (LeftEdge: xLoc; BottomEdge: yLoc;
         RightEdge: xLoc+Width; TopEdge: yLoc+Height)
    TextLayoutFlushRight()
    Type (Text: x+" ")
    TextEditExit()
endfor
xLoc = (1.1*Margin)
for (y; 0; y < 16; y + 1)
    yLoc = (16 * Height) - (y * Height) + Margin + (Height/2)
    TextBoxCreate ()
    AddTextBox
        (LeftEdge: xLoc; BottomEdge: yLoc;
         RightEdge: xLoc+Width; TopEdge: yLoc+Height)
    TextLayoutFlushRight()
    Type (Text: y*16)
    TextEditExit()
endfor
BeginLine ()
ObjectSetLineWidth (LineWidth: 80w)

```

```
AddPolyLine (FillAndFrame: OpenFrame!; PointCount: 2;
              {PointX: 600w; PointY: 8676w; PointX: 12600w; PointY: 8676w})
UnselectAllObjects()
```

Return

Label(SelectPalette)

```
vPalettePath := EnvPaths(Palettes!)
DialogDefine("GetPaletteDlg"; 50; 50; 182; 62; Percent!; "Retreive Palette")
DialogAddText("GetPaletteDlg"; "SText1"; 6; 10; 60; 10; 1; "Palette Filename:")
DialogAddFilenameBox("GetPaletteDlg"; "FileName"; 5; 19; 117; 14; FilesAndDirs!; vPaletteName; vPalettePath; "*.prc")
DialogAddPushButton("GetPaletteDlg"; "OK_Btn"; 133; 19; 36; 13; 0; " OK ")
DialogDisplay("GetPaletteDlg"; "FileName")
DialogDestroy("GetPaletteDlg")
if (vPaletteName = "")
    Quit
else
    RetrieveColorPalette(vPaletteName)
    vStringLen := StrLen(vPaletteName)
    vDirLen := StrLen(vPalettePath)
    vTotalLen := vStringLen - vDirLen -4 // 4 for the extension
    vPaletteName := SubStr(vPaletteName;vDirLen + 1;vTotalLen)
```

endif

Return