

```

//*****
//* MACRO: SHW2WPG.WCM
//* PURPOSE: Save all the slides in your slide show to WPG's
//*****
APPLICATION (Presentations; "Presentations"; Default; "EN")

vShow:=""
vDir:=""
vMin:=0
vStepNum:=0
vStepStay:=0
vWhere:=EnvPaths(Presentations!)
vDone:=0

    DIALOGDEFINE(100;50;50;200;125;Percent!|Cancel!;"SHW to WPG")
    DIALOGADDTTEXT(100;101;7;3;200;15;1;"&Slide show to convert:")
    DIALOGADDFILENAMEBOX(100;102;5;12;183;16;FilesandDirs!;vShow;vWhere;"Presentations 7 slide show (*.shw)")
    DIALOGADDTTEXT(100;103;7;32;200;15;1;"S&ave .wpg files to:")
    DIALOGADDFILENAMEBOX(100;104;5;42;183;16;DirOnly!;vDir;vWhere;"Folders only")
    DIALOGADDCHECKBOX(100; 105; 8; 62; 175; 10; "Minimi&ze Corel Presentations for faster conversion"; vMin)
    DIALOGADDTTEXT(100;106; 8; 80; 185;15;1;"WPG files will be named according to the slide title. If the slide has no title, the slide
number will be used.")
    DIALOGADDPUSHBUTTON(100; "CANCELButton"; 139; 107; 50; 13; 1; " Cancel ")
    DIALOGADDPUSHBUTTON(100; "CONVERTButton"; 80; 107; 50; 13; 1; " &Convert ")
    DIALOGLOAD(100; "Presentations")
    RegionShowWindow("100.CancelBttn"; Hide!)
    DIALOGSHOW(100;;MainLoop)
    CallBackWait()
    DialogDestroy(100)
    vDirLen := STRLEN(vDir)
    vBackSlashSource := SUBCHAR(vDir;vDirLen ;1)
    IF(vBackSlashSource != "\")
        vDir := vDir + "\"
        vDirLen := vDirLen + 1
    ENDIF
    IF(vMin = 1)
        PRActivate ()
        ApplicationMinimize ()
    ENDIF
    FileOpen (vShow; Yes!)
    Call(SaveShows)
    FileClose (No!)
    IF(vMin = 1)
        ApplicationRestore ()
        PRActivate ()
    ENDIF
QUIT

LABEL (MainLoop)
    if(MainLoop[5]=274)
        DialogDestroy(100)
        QUIT
    endif
    if(MainLoop[5]=273)
        switch(MainLoop[3])
            caseof"CONVERTButton":
                vShow = RegionGetWindowText("100.102")
                vDir = RegionGetWindowText("100.104")
                IF((vDir = "")OR(vShow = ""))
                    MessageBox(vStatus;"No File or Path Entered";"Please make sure a filename and path have been specified."; OK! |
IconStop!)
                ELSE

```

```

    IF(DoesFileExist(vShow)!=TRUE)
        MessageBox(vStatus;"File Not Found";"File entered could not be found, enter in existing file"; OK! | IconStop!)
    ELSE
        IF(DOESDIRECTORYEXIST(vDir)!=TRUE)
            MessageBox(vYesNo;"Directory Not Found";"Directory entered could not be found, do you wish to create
directory: "+ vDir; YesNo! | IconQuestion!)
            IF(vYesNo = 6) // If yes
                CREATEDIRECTORY(vDir; 0)
                vDone:=1
            ELSE
                ENDIF
        ELSE
            vDone:=1
        ENDIF
    ENDIF
    IF(vDone = 1)
        DialogDismiss("100";"CONVERTButton")
        CallBackResume()
        Return
    ENDIF
    caseof"CancelBttn":
        DialogDestroy(100)
        QUIT
    caseof"CANCELButton":
        DialogDestroy(100)
        QUIT
    endswitch
endif
RETURN

```

```

LABEL(SaveShows)
    WaitMsgInit ()
    vSlideNum:=EnvNumberOfSlides()
    vStepStay:=(100 / vSlideNum)
    Declare vSlideTitles[vSlideNum]
    Declare vInvalidFileChars[17]
    SlideShowOutline()
    ShowSlide(1)

```

```

vInvalidFileChars[1] = "\"
vInvalidFileChars[2] = "/"
vInvalidFileChars[3] = ":"
vInvalidFileChars[4] = "*"
vInvalidFileChars[5] = "?"
vInvalidFileChars[6] = "\""
vInvalidFileChars[7] = "\""
vInvalidFileChars[8] = "<"
vInvalidFileChars[9] = ">"
vInvalidFileChars[10] = "|"
vInvalidFileChars[11] = "@"
vInvalidFileChars[12] = "%"
vInvalidFileChars[13] = "^"
vInvalidFileChars[14] = "&"
vInvalidFileChars[15] = "("
vInvalidFileChars[16] = ")"
vInvalidFileChars[17] = "!"

```

```

for(x; 1; x<(vSlideNum+1); x+1)
    WaitMsgDisplay ("Converting slides..."; vStepNum)
    vStepNum:=(vStepNum + vStepStay)

```

```

vHoldSlideTitle = EnvSlideTitle()
for (xi; 1; xi<18; xi+1)
    vHoldSlideTitle := StrTransform (String:vHoldSlideTitle; FromChars:vInvalidFileChars[xi]; ToChars:"_"; Options:All!)
endfor
vSlideTitles[x] := vHoldSlideTitle

if(vSlideTitles[x] = "")
    vSlideTitles[x]:= "slide"+x
else
    if(x != 1)
        for(y; 1; y<(x); y+1)
            if(vSlideTitles[x] = vSlideTitles[y])
                vSlideTitles[x]:=EnvSlideTitle()+x
            endif
        endfor
    endif
endif
FileSaveAs (vDir+vSlideTitles[x]+".wpg"; WPGraphic_70!; TRUE; FALSE)
SlideNext()
endfor
WaitMsgDisplay ("Converting slides..."; 100)
WaitMsgHide ()
WaitMsgDestroy ()
SlideShowSlide()
RETURN

//*****
// Wait Message Procedures
//*****
PROCEDURE WaitMsgInit ()
    DialogDefine ("WaitMsg"; 50; 50; 150; 50; 16; "Please Wait")
    DialogAddText ("WaitMsg"; "WaitText"; 8; 10; 144; 16; 1; "")
    DialogAddProgress("WaitMsg"; "WaitBar"; 10; 30; 130; 15)
    DialogLoad ("WaitMsg"; "Presentations")
    Return
ENDPROC
PROCEDURE WaitMsgDisplay (Text; Number)
    RegionSetWindowText ("WaitMsg" + ".WaitText"; Text)
    RegionSetProgressPercent("WaitMsg" + ".WaitBar"; Number)
    DialogShow ("WaitMsg";; WaitDlgCallBack)
ENDPROC
PROCEDURE WaitDlgCallBack ()
    If (WaitDlgCallBack[5] = 274 And WaitDlgCallBack[6] = 61536)
        Assert (CancelCondition!)
    EndIf
ENDPROC
PROCEDURE WaitMsgHide ()
    DialogDismiss("WaitMsg"; "WaitText")
ENDPROC
PROCEDURE WaitMsgDestroy ()
    DialogDestroy ("WaitMsg")
ENDPROC
//*****

```