

```
//*****  
//  
// NAME:          WPCH_OT1.WPM  
//  
// PURPOSE:    Coach on outline-create  
//  
// AUTHOR:    Aaron Grant, WordPerfect Corp.  
//  
//*****
```

```
// Revision History  
//  
//
```

```
Error(Off!)  
Global(ScrnMode;MsgAdj;DefTitle;CancelMsg)  
Global(xposS1;xposS2;yposS1;FakeBtns)  
Global(StepNum;StepText;SaveScrn)  
Global(Element;HotElement)
```

```
CancelMsg=0  
select[0]=0  
xpos=0  
ypos=0
```

```
DefTitle="Coach"  
ScrnMode=?ScreenHigh-24
```

```
StepNum=0           // Initialize step number  
StepText=""         // Initialize step text
```

```
Use("wpchlibr.wpm") // Shared library file  
ScreenSetup()  
ColorScheme()  
MsgAdj=DlgMessage()
```

```

//*****
//
//  MAIN PROGRAM: Coach
//
//*****
Cancel(Off!)
Call(WelcomeDlg)           //Welcome Dlg
Label(Main)
steps={  "Creating a standard outline";
        "Modifying a standard outline";
        "Selecting a different outline style";
        }
    Call(OverviewDlg)      // Overview Dialog
If(Choice=1)
    Label(Top)
    Call(PosCursor)
    WindowMaximize
    Call(OutlineDlg)
    WindowMaximize

    If(?Dialog=479)      //If Ctrl-f5 was used
        Call(OutlineDlg2) // walk through of choosing 1 begin new outline
    EndIF
    Call(OutStyList)
    Call(TypeOutline)
    Call(PosCurEnd)
    Call(EndOutline)
    If(?Dialog=479)      //If Ctrl-f5 was used
        Call(EndOutline2) // walk through of choosing 5 end outline
    EndIF

    Message(&tbox;320)      //Do you want to edit?
    Result=MessageBox(DefTitle;&tbox;3)
    If(Result=1)
        Call(EditOutline)
        Message(&tbox;325)
        Result=MessageBox(DefTitle;&tbox;3)
        If(Result=1)
            Call(COutline)
        EndIf
    EndIf
    Display(Off!)
    Display(On!)
Else
    If(Choice=2)
        Call(FindOut)

```

```
    Call(EditOutline)
    Message(&tbox;325)
    Result=MessageBox(DefTitle;&tbox;3)
    If(Result=1)
        Call(COutline)
    EndIf
Else
    If(Choice=3)
        Call(FindOut)
        Call(COutline)
    EndIf
EndIf
EndIf
WindowMaximize
message(&tbox;800)
Result=Messagebox(DefTitle;&tbox;3)
If(Result=1)
    Go(Main)
EndIf
QuitCoach()
//*****
//  END MAIN PROGRAM
//*****
```

```
Label(FindOut)
  Label(Srch)
  result=0
  junk=OpenDocument()
  Display(Off!)
  PosDocVeryTop
  OnNotFound Call(NoOutline)
  SearchString("[Outline]")
  SearchNext()
  Display(On!)
  If(Result=1)
    Go(top)
  Else
    If(Result=2)
      Go(Srch)
    Else
      If(Result=-1)
        CancelCoach()
        Go(Srch)
      EndIf
    EndIf
  EndIf
EndIf
Return
```

```
Label(NoOutline)
Message(&tbox;2010)
Result=MessageBox(DefTitle;&tbox;4)
Return
```

```
//*****  
//Subroutine: Welcome  
//*****  
Label(WelcomeDlg)  
Message(&tbox;5)  
Result=Welcome(DefTitle;&tbox)  
If(Result=-1)  
    QuitCoach()  
EndIf  
Return
```

```

//*****
//
//  SUBROUTINE: OverviewDlg
//
//*****
Label(OverviewDlg)
  w=63  h=9
  DlgCreate(Result;DefTitle;DlgNoCancel!+DlgNoOK!+DlgExit!;;;w;h)
  ForNext(x;1;steps[0])
  Message(&tbox;10)
  DlgControl(CtrlRadioButton!;
    Var[x];steps[x];4;x+tbox[0];w-1;1)
  Endfor
  ForNext(x;1;tbox[0])
    DlgControl(CtrlLabel!;;tbox[x];2;x;w-1;1)
  Endfor
  DlgControl (CtrlPushButton!;;"Quit...[Bold On]Esc[Bold Off]";StyCancel!;w-14;h-2)
  Var[1]=0 Var[2]=0 Var[3]=0
  DlgEnd

  If(Result=-1)
    CancelCoach()
    Go(Main)
  EndIf
  Choice=Result
Return

```

```
//Subroutine: PosCursor  
Label(PosCursor)  
Message(&tbox;50)  
FakeIt(DefTitle;&tbox;3;1;0)  
PosCharNext  
PosCharPrev  
keypress={-8154;}  
chkcode={-1;}  
Result=DoEdit(1;&chkcode;&keypress)  
Return
```



```

//*****
//
// SUBROUTINE: OutlineDlg
//
// PURPOSE: Walk user through getting to the outline dlg
//
//*****
Label(OutlineDlg)
    Message(&tbox;100)
    hbox1={ "Choose [Bold On]Tools.";
            }
    hbox2={ "Choose [Bold On]Outline.";
            }
    hbox3={ "Choose [Bold On]Begin New Outline.";
            }
    ToolsMenu(&valid)
    allow={ 1603; //Begin new Outline menu option
           }

    exallow={ 0; // Allow execute
             }

    accel={ 111; //(O)utline
           }

    exaccel={ 1; // Allow execute
             }

    subval={ 1602;
             1603; // Outline options
             1604;
             1605;
             1606;
             1607;
             1608;
             1609;
             1610;
             1611;
             1612;
             1613;
             1614;
             1615;
           }
    subacl={ 98; // (B)egin new Outline option
            }
    exsubacl={ 0; // Allow execute
              }

```

```

    }
    subs = {
        1601; // Outline
    }
    sbox = {-1;
    }
    funkey = {
        -8120; // Ctrl+F5
    }

```

Repeat

```

Result=PostIt(DefTitle;&tbox;0;12;0;2;0)
MenuActive=0
Show=HotSpot(0;0;62;0;-529;&funkey;1)

```

Switch(show)

```

CaseOf 1: // Show menu
    sbox={"Choose [Bold On]Begin New Outline.";}
    Result=Postit(DefTitle;&tbox;0;12;0;3;0)
    ShowMenu(DefTitle;&sbox;3;6-ScrnMode;4;2;2;1)

```

CaseOf 1592: // Tools menu ID

```

CaseOf -529: // Accel menu alt-t
    Result=Postit(DefTitle;&tbox;0;12;0;3;0)
    PullDownDlg
    Type("t")

```

CaseOf -8120: // Function key

```

    DlgInput(On!)
    DisplayRewrite
    OutlineDlg

```

CaseOf -8085: // Alt+=

```

    sbox={-1;}
    Result=PostIt(DefTitle;&tbox;0;12;0;3;0)
    MenuActive=MenuBarControl("",&hbox1;&sbox;0;3-ScrnMode;116;&funkey)

```

DEFAULT: // Default

```

    sbox={-1;}
    MenuActive=MenuBarControl("",&hbox1;&sbox;0;3-ScrnMode;116;&funkey)

```

EndSwitch

If(MenuActive<>-1)

If(Show <> -8120 and MenuActive<>-8120) //If function key was not pressed

```

Result=MenuControl(&hbox1;&hbox2;&hbox3;&valid;&allow;&exallow;&accel;&exaccel;&subval;&subacl;&exsub
bacl;&subs;&funkey)
    MenuActive=Result

```

```
        CancelMenus()
    EndIf
EndIf
If(menuactive=-1)
    Result=PostIt(DefTitle;&tbox;0;12;0;2;0)
EndIf
Until(MenuActive<>-1)
DisplayRewrite
CancelMenus()
OutlineBeginNewDlg
Return
```

```

//*****
// LABEL: Outline Dlg 2
//
// PURPOSE: Walk user through choosing 1 begin new outline
//
//*****
Label(OutlineDlg2)
message(&tbox;500)
SaveScrn=0
While(?Dialog=479)
    Switch(?DialogControl)

        CaseOf 1131:
            info={1131;479;0;"";10;22;0;"";1;0}
            x1={7;}
            x2={29;}
            y1={6;}
            y2={6;}
            Hotkey={49;}
            HotPnt={1;}
            Accel={49;98;}
            aclpnt={1;1}
            message(&tbox;500)
            Helpmsg="Choose [Bold On]Begin New Outline."
            hbox=""
            Result=ButtonControl(&info;&x1;&x2;&y1;&y2;&hotkey;
                                &hotpnt;&accel;&aclpnt;&tbox;&helpmsg)

        Default:
            info={?DialogControl;479;1;"";10;22;0;"";0;0}
            x1={7;}
            x2={29;}
            y1={6;}
            y2={6;}
            Hotkey={49;}
            HotPnt={1;}
            accel={49;98;}
            aclpnt={1;1;}
            Helpmsg="Choose [Bold On]Begin New Outline."
            allow={ 1131;}
            hbox=""
            Result=InvalidControl(&info;&x1;&x2;&y1;&y2;&hotkey;
                                &hotpnt;&accel;&aclpnt;&tbox;&helpmsg;&allow)
    EndSwitch

EndWhile
Return

```

```

//*****
// LABEL : OutStyList
// PURPOSE: Walk user through choosing 1 select
//*****
Label(OutStyList)
While(?Dialog<>-1)
  If(Result="Hint")
    message(&hbox;2000)
    Void=Hint(&hbox;-99;-99;3)
  EndIf
  CurDlg=?Dialog
  Switch(?Dialog)
  CaseOf 465:          //Main Outline sty dlg
    Message(&tbox;190) //Choose select
    Result=PostIt(DefTitle;&tbox;12;21;0;4;0)

  CaseOf 425:          //chose Create
    Message(&tbox;120) //Choose select
    Result=PostIt(DefTitle;&tbox;16;18;0;0;0)

  CaseOf 190:
    Message(&tbox;130) //Chose edit
    Result=PostIt(DefTitle;&tbox;60;9;0;0;0)

  CaseOf 139:
    Message(&tbox;150) //Chose copy
    Result=PostIt(DefTitle;&tbox;10;19;0;0;0)

  CaseOf 499:
    Message(&tbox;160) //Chose options
    Result=PostIt(DefTitle;&tbox;10;19;0;0;0)

  CaseOf 454:
    Message(&tbox;170) //Chose Save
    Result=PostIt(DefTitle;&tbox;10;19;0;0;0)

  CaseOf 453:
    Message(&tbox;180) //Chose retrieve
    Result=PostIt(DefTitle;&tbox;10;19;0;0;0)

  EndSwitch

  While(CurDlg=?Dialog)
    Look(key)
    If(key=-768)
      If(?MouseY=yposS1)

```

```
    If(?MouseX>xposH1)
        If(?MouseX<xposH2)
            Result="Hint"
            BREAK
        EndIf
    EndIf
EndIF
If(?MouseX>xposQ1)
    If(?MouseX<xposQ2)
        CancelCoach()
        key=0
    EndIf
EndIf
EndIf
EndIf
If(key=-8158)
    Result="Hint"
    BREAK
EndIf
If(key=-8160)
    CancelCoach()
    key=0
EndIf
    Type(ntoc(key))
EndWhile
EndWhile

If(?LeftCode=-8953 or ?leftCode=-11213)
Else
    Message(&tbox;1000)
    DisplayRewrite
    Result=MessageBox(DefTitle;&tbox;3)
    If(Result=2)
        CancelCoach()
    EndIf
    Go(Top)
EndIf
Return
```

```

//*****
// LABEL : TypeOutline
// PURPOSE: Walk users through typing levels of headings
//*****
Label(TypeOutline)
Message(&tbox;240) //Hot to do first level
keypress={-8182;-8154}
FakeIt(DefTitle;&tbox;1;1;0)
PosCharPrevious
PosCharNext
chkcode={-8953;}
Repeat
    Result=DoEdit(1;&chkcode;&keypress)
    If(Result=-8953)
        Message(&tbox;260)
        Junk=Hint(&tbox;-99;-99;3)
        PosCharPrev
        PosCharNext
    EndIf
Until(Result=-8154 or Result=1 or Result=-8182)
Message(&tbox;250) //How to do second level
keypress={-8154;}
FakeIt(DefTitle;&tbox;3;1;1)
PosCharPrevious
PosCharNext
Repeat
    Result=DoEdit(1;&chkcode;&keypress)
    If(Result=-8953)
        Message(&tbox;260)
        Junk=Hint(&tbox;-99;-99;0)
        PosCharPrev
        PosCharNext
    EndIf
Until(Result=-8154 or Result=1)
WindowMaximize
DisplayRewrite
Return

```

```
//*****  
//  
// Label: PosCurEnd  
//  
//*****  
Label(PosCurEnd)  
Message(&tbox;310)  
FakeIt(DefTitle;&tbox;3;1;0)  
PosCharPrev  
PosCharNext  
keypress={-8154;}  
chkcode={-1;}  
Repeat  
    Result=DoEdit(1;&chkcode;&keypress)  
Until(Result=1 or Result=-8182 or Result=-8154)  
WindowMaximize  
Return
```



```

//*****
//
// LABEL: EndOutline
//
// PURPOSE: Walk user through ending the outline
//
//*****
Label(EndOutline)
Message(&tbox;101)
    hbox1={ "Choose [Bold On]Tools.";
            }
    hbox2={ "Choose [Bold On]Outline.";
            }
    hbox3={ "Choose [Bold On]End Outline.";
            }
ToolsMenu(&valid)

allow={      1605;    //End Outline menu option
        }

exallow={0;}

accel={      111;    //(O)utline
        }
mnuexecute={0;    // Allow execute
            }

exaccel={1;}

subval={      1602;
            1603; // Outline options
            1604;
            1605;
            1606;
            1607;
            1608;
            1609;
            1610;
            1611;
            1612;
            1613;
            1614;
            1615;

        }

```

```
exsubacl={-1;}
```

```
subacl={      101;  // (E)nd Outline option  
        }
```

```
subs ={      1601; // Outline  
        }
```

```
hbox={      "Choose [Bold On]Tools."  
          ""  
        }
```

```
sbox={-1;}
```

```
funkey={      -8120; // Ctrl+F5  
        }
```

Repeat

```
Result=PostIt(DefTitle;&tbox;0;12;0;2;0)  
MenuActive=0  
Show=HotSpot(0;0;62;0;-529;&funkey;1)
```

Switch(show)

CaseOf 1: // Show menu

```
sbox={"Choose [Bold On]End Outline.";}  
Result=Postit(DefTitle;&tbox;0;12;0;3;0)  
ShowMenu(DefTitle;&sbox;11;8-ScrnMode;4;2;4;1)
```

CaseOf 1592: // Tools menu ID

CaseOf -529: // Accel menu alt-t

```
PullDownDlg  
Type("t")
```

CaseOf -8120: // Function key

```
DlgInput(On!)  
DisplayRewrite  
OutlineDlg
```

CaseOf -8085: // Alt+=

```
sbox={-1;}  
Result=PostIt(DefTitle;&tbox;0;12;0;3;1)  
MenuActive=MenuBarControl("",&hbox;&sbox;3;3-ScrnMode;116;&funkey)
```

DEFAULT: // Default

```
sbox={-1;}  
MenuActive=MenuBarControl("",&hbox;&sbox;3;3-ScrnMode;116;&funkey)
```

EndSwitch

```
If(MenuActive<>-1)
  If(Show <> -8120 and MenuActive<>-8120) //If function key was not pressed
Result=MenuControl(&hbox1;&hbox2;&hbox3;&valid;&allow;&exallow;&accel;&exaccel;&subval;&subacl;&exsu
bacl;&subs;&funkey)
  MenuActive=Result
  CancelMenus()
  EndIf
EndIf
If(menuactive=-1)
  Result=PostIt(DefTitle;&tbox;0;12;0;2;0)
EndIf
Until(MenuActive<>-1)
DisplayRewrite
CancelMenus()
OutlineEnd
Return
```

```

//*****
//
// Label: EndOutline2
//
// Purpose: Walk user through choosing 5. end outline
//
//*****
Label(EndOutline2)
message(&tbox;510)
SaveScrn=0
While(?Dialog=479)
    Switch(?DialogControl)

        CaseOf 1447:
            info={1447;479;0;DefTitle;10;22;0;DefTitle;1;0}
            x1={7;}
            x2={20;}
            y1={10;}
            y2={10;}
            Hotkey={53;}
            HotPnt={1;}
            Accel={53;101;}
            aclpnt={1;1}
            message(&tbox;510)
            Helpmsg="Choose [Bold On]End Outline."
            hbox=""
            Result=ButtonControl(&info;&x1;&x2;&y1;&y2;&hotkey;
                                &hotpnt;&accel;&aclpnt;&tbox;&helpmsg)

        Default:
            info={?DialogControl;479;1;DefTitle;10;22;0;DefTitle;0;0}
            x1={7;}
            x2={20;}
            y1={10;}
            y2={10;}
            Hotkey={53;}
            HotPnt={1;}
            Accel={53;101;}
            aclpnt={1;1}
            message(&tbox;510)
            Helpmsg="Choose [Bold On]End Outline."
            hbox=""
            Result=InvalidControl(&info;&x1;&x2;&y1;&y2;&hotkey;
                                &hotpnt;&accel;&aclpnt;&tbox;&helpmsg;&allow)

    EndSwitch
EndWhile
Return

```

```
//*****  
//  
// Label: EditOutline  
//  
//*****  
Label(EditOutline)  
Call(EditOverview)  
If(Var[1]=1)  
    Call(EditInsert)  
EndIf  
If(Var[2]=1)  
    Call(EditAdjust)  
EndIf  
WindowMaximize  
Return
```

```

//*****
//
// Label(EditOverview)
//
//*****
Label(EditOverview)
steps={  "Insert a new heading";
         "Change the level of an existing heading"
        }
w=55 h=11
DlgCreate(Result;DefTitle;DlgNoCancel!+DlgNoOK!;;;w;h)
Message(&tbox;600)
ForNext(x;1;tbox[0])
    DlgControl(CtrlLabel!;;tbox[x];;2;x;w-1;1)
Endfor
ForNext(x;1;steps[0])
    DlgControl(CtrlCheckBox!;
    Var[x];steps[x];;4;x+tbox[0];w-1;1)
Endfor
DlgControl(CtrlLabel!;;"Select one or both activities and choose OK.";;2;h-4)
DlgControl(CtrlPushButton!;;"Quit...[Bold On]Esc[Bold Off]";StyCancel!;w-14;h-2)
DlgControl(CtrlPushButton!;;" OK ";StyOk!+StyInitial!;w-23;h-2)
Var[1]=0 Var[2]=0
DlgEnd

If(Result=-1)
    CancelCoach()
    Go(EditOutline)
EndIf

If(Var[1]=0 and Var[2]=0)
    Message(&tbox;1010)
    Result=MessageBox(DefTitle;&tbox;5)
    If(Result=1)
        QuitCoach()
    Else
        Go(EditOutline)
    EndIf
EndIf
Return

```

```
Label(EditInsert)
message(&tbox;610)
keypress={-8154;}
FakeIt(DefTitle;&tbox;3;1;1)
PosCharPrevious
PosCharNext
chkcode={-1;}
Result=DoEdit(1;&chkcode;&keypress)
Return
```

```
Label(EditAdjust)
message(&tbox;620)
keypress={-8154;}
FakeIt(DefTitle;&tbox;3;1;0)
PosCharPrevious
PosCharNext
chkcode={-1;}
Result=DoEdit(1;&chkcode;&keypress)
Return
```



```
Label(COutline)
Call(CWelcome)
Call(CPos)
Label(CAgain)
Call(CGetDlg)
If(?Dialog=479)
    Call(CGetDlg2)
EndIf
Call(CStyDlg)

Return
```

```
Label(CWelcome)
Message(&tbox;400)
Result=Welcome(DefTitle;&tbox) //Welcome to convert
If(Result=-1)
    CancelCoach()
    Go(COutline)
EndIf
Return
```

```
Label(Cpos)
Message(&tbox;410)           //Pos your cursor
FakeIt(DefTitle;&tbox;3;1;0)
PosCharNext
PosCharPrev
keypress={-8154;}
chkcode={-1;}
Repeat
    Result=DoEdit(1;&chkcode;&keypress)
Until(Result=1 or Result=-8154)
WindowMaximize
Return
```

```

Label(CGetDlg)
    Message(&tbox;420)
    hbox1={ "Choose [Bold On]Tools.";
    }
    hbox2={ "Choose [Bold On]Outline.";
    }
    hbox3={ "Choose [Bold On]Outline Style.";
    }
    ToolsMenu(&valid)
    allow={ 1604; //Outline Style menu option
    }

    exallow={ 0; // Allow execute
    }

    accel={ 111; //(O)utline
    }

    exaccel={ 1; // Allow execute
    }

    subval={ 1602;
    1603; // Outline options
    1605;
    1606;
    1607;
    1608;
    1609;
    1610;
    1611;
    1612;
    1613;
    1614;
    1615;
    }
    subacl={ 111; // (O)utline style option
    }
    exsubacl={ 0; // Allow execute
    }
    subs = { 1601; // Outline
    }
    sbox={ -1;
    }
    funkey={ -8120; // Ctrl+F5
    }

```

Repeat

```
Result=PostIt(DefTitle;&tbox;0;12;0;2;0)
MenuActive=0
Show=HotSpot(0;0;62;0;-529;&funkey;1)
```

```
Switch(show)
```

```
CaseOf 1: // Show menu
  sbox={"Choose [Bold On]Outline Style.";}
  Result=Postit(DefTitle;&tbox;0;12;0;3;0)
  ShowMenu(DefTitle;&sbox;3;6-ScrnMode;4;2;3;1)
```

```
CaseOf 1592: // Tools menu ID
```

```
CaseOf -529: // Accel menu alt-t
  Result=Postit(DefTitle;&tbox;0;12;0;3;0)
  PullDownDlg
  Type("t")
```

```
CaseOf -8120: // Function key
  DlgInput(On!)
  DisplayRewrite
  OutlineDlg
```

```
CaseOf -8085: // Alt+=
  sbox={-1;}
  Result=PostIt(DefTitle;&tbox;0;12;0;3;0)
  MenuActive=MenuBarControl("",&hbox1;&sbox;0;3-ScrnMode;116;&funkey)
```

```
DEFAULT: // Default
  sbox={-1;}
  MenuActive=MenuBarControl("",&hbox1;&sbox;0;3-ScrnMode;116;&funkey)
```

```
EndSwitch
```

```
If(MenuActive<>-1)
```

```
  If(Show <> -8120 and MenuActive<>-8120) //If function key was not pressed
```

```
Result=MenuControl(&hbox1;&hbox2;&hbox3;&valid;&allow;&exallow;&accel;&exaccel;&subval;&subacl;&exsubacl;&subs;&funkey)
```

```
  MenuActive=Result
```

```
  CancelMenus()
```

```
  EndIf
```

```
EndIf
```

```
If(menuactive=-1)
```

```
  Result=PostIt(DefTitle;&tbox;0;12;0;2;0)
```

```
EndIf
```

```
Until(MenuActive<>-1)
```

DisplayRewrite
CancelMenus()
OutlineStyleDlg
Return

```

Label(CGetDlg2)
message(&tbox;515)
SaveScrn=0
While(?Dialog=479)
    Switch(?DialogControl)

        CaseOf 1445:
            info={1445;479;0;"";10;-99;0;"";1;0}
            x1={7;}
            x2={25;}
            y1={8;}
            y2={8;}
            Hotkey={51;}
            HotPnt={1;}
            Accel={51;111;}
            aclpnt={1;1}
            message(&tbox;515)
            Helpmsg="Choose [Bold On]Outline Style."
            hbox=""
            Result=ButtonControl(&info;&x1;&x2;&y1;&y2;&hotkey;
                                &hotpnt;&accel;&aclpnt;&tbox;&helpmsg)

        Default:
            info={?DialogControl;479;1;"";10;22;0;"";0;0}
            x1={7;}
            x2={25;}
            y1={8;}
            y2={8;}
            Hotkey={51;}
            HotPnt={1;}
            Accel={51;111;}
            aclpnt={1;1}
            message(&tbox;515)
            Helpmsg="Choose [Bold On]Outline Style."
            allow={ 1131;}
            hbox=""
            Result=InvalidControl(&info;&x1;&x2;&y1;&y2;&hotkey;
                                &hotpnt;&accel;&aclpnt;&tbox;&helpmsg;&allow)
    EndSwitch

EndWhile
Return

```

```

Label(CStyDlg)
While(?Dialog<>-1)
  If(Result="Hint")
    message(&hbox;2000)
    Void=Hint(&hbox;-99;-99;3)
  EndIf
  CurDlg=?Dialog
  Switch(?Dialog)
  CaseOf 465:           //Main Outline sty dlg
    Message(&tbox;195) //Choose select
    Result=PostIt(DefTitle;&tbox;12;21;0;4;0)

  CaseOf 425:          //chose Create
    Message(&tbox;120) //Choose select
    Result=PostIt(DefTitle;&tbox;16;18;0;0;0)

  CaseOf 190:
    Message(&tbox;130) //Chose edit
    Result=PostIt(DefTitle;&tbox;60;9;0;0;0)

  CaseOf 139:
    Message(&tbox;150) //Chose copy
    Result=PostIt(DefTitle;&tbox;10;19;0;0;0)

  CaseOf 499:
    Message(&tbox;160) //Chose options
    Result=PostIt(DefTitle;&tbox;10;19;0;0;0)

  CaseOf 454:
    Message(&tbox;170) //Chose Save
    Result=PostIt(DefTitle;&tbox;10;19;0;0;0)

  CaseOf 453:
    Message(&tbox;180) //Chose retrieve
    Result=PostIt(DefTitle;&tbox;10;19;0;0;0)
  EndSwitch

While(CurDlg=?Dialog)
  Look(key)
  If(key=-768)
    If(?MouseY=yposS1)
      If(?MouseX>xposH1)
        If(?MouseX<xposH2)
          Result="Hint"
          BREAK
        EndIf
      EndIf
    EndIf
  EndIf

```



```

        EndIf
        If(?MouseX>xposQ1)
            If(?MouseX<xposQ2)
                cancelmsg=0
                CancelCoach()
                key=0
            EndIf
        EndIf
    EndIf
EndIf
If(key=-8158) //F1 key
    Result="Hint"
    BREAK
EndIf
If(key=-8160) //Esc key
    cancelmsg=0
    CancelCoach()
    key=0
EndIf
CurSty=?Entry
Type(ntoc(key))
EndWhile
EndWhile

If(CurSty=0)
    Message(&tbox;1025)
    Result=MessageBox(DefTitle;&tbox;3)
Else
    Global(CurSty)
    Call(ExplnSty)
EndIf
If(Result=2)
    QuitCoach()
EndIf
Go(CAgain)
Return

```

```
Label(ExplnSty)
If(CurSty="Headings")
tbox={ "You chose the Headings style. With this style";
      "selected, your outline headings appear as document";
      "headings—chapter titles, section titles, subsection";
      "titles, and so forth. In addition, they have been";
      "automatically marked for inclusion in a table of";
      "contents.";
      """,
      "To transform your outline into a full document,";
      "fill in the text between the headings.";
      """,
      "Would you like to select a different style?";
      """,
      }
Else
tbox={ "You chose the " + CurSty + " outline style.";
      """,
      "Would you like to select a different";
      "style?";
      """,
      }
EndIf
DisplayRewrite
Result=MessageBox(DefTitle;&tbox;3)
Return
```

```

//*****
//
// PROCEDURE: Message
//
//*****
PROCEDURE Message(&tbox;Msg)

Switch(Msg)

CaseOf 5:
tbox={ "";
    "      [Bold On]Creating an Outline[Bold Off]";
    """,
    "You can easily create and modify outlines as well";
    "as convert their headings to document headings.";
    "These outlines can appear in a number of forms, but";
    "the most common is the standard outline similar to";
    "the following:";
    """,
    "I.  Heading level one";
    "  A.  Heading level two";
    "  B.  Heading level two";
    "    1.  Heading level three";
    "    2.  Heading level three";
}

```

```

CaseOf 10:
tbox={ "Choose the activity that you are interested in.";
    """,
}

```

```

CaseOf 50:
tbox={ "► Place the cursor where you want the outline to go.";
    """,
}

```

```

CaseOf 100:
tbox={ "To start your outline,";
    """,
    "► Choose [Bold On]Tools[Bold Off],";
    " then [Bold On]Outline[Bold Off],";
    " then [Bold On]Begin New Outline[Bold Off].";
    " -or-";
    "► Press [Bold On]Ctrl+F5[Bold Off].";
}

```

```
    "";  
  }
```

CaseOf 101:

```
tbox={ "To end your outline,";  
    "";  
    "► Choose [Bold On]Tools[Bold Off],";  
    " then [Bold On]Outline[Bold Off],";  
    " then [Bold On]End Outline[Bold Off].";  
    " -or-";  
    "► Press [Bold On]Ctrl+F5[Bold Off].";  
    "";  
  }
```

CaseOf 120:

```
tbox={ "You chose Create. This option is";  
    "used to create a custom outline style.";  
    "► Choose [Bold On]Cancel[Bold Off] to return to the previous";  
    " dialog.";  
  }
```

CaseOf 130:

```
tbox={ "You chose Edit. This";  
    "option is used to edit";  
    "an existing outline";  
    "style.";  
    "► Choose [Bold On]Cancel[Bold Off] to";  
    " return to the previous";  
    " dialog.";  
  }
```

CaseOf 150:

```
tbox={ "You chose Copy. This is used to copy outline styles";  
    "to a different library or to a document.";  
    "► Choose [Bold On]Cancel[Bold Off] to return to the previous dialog.";  
  }
```

CaseOf 160:

```
tbox={ "You chose Options. This is used to change certain";  
    "outline style attributes.";  
    "► Choose [Bold On]Cancel[Bold Off] to return to the previous dialog.";  
  }
```

CaseOf 170:

```
tbox={ "You chose Save. This is used to save the current document";  
    "styles as a library that can be retrieved into other documents.";
```

```
    "► Choose [Bold On]Cancel[Bold Off] to return to the previous dialog.";
}
```

CaseOf 180:

```
tbox={ "You chose Retrieve. This is used to retrieve a style library";
      "into the current document.";
      "► Choose [Bold On]Cancel[Bold Off] to return to the previous dialog.";
}
```

CaseOf 190:

```
tbox={ "► Highlight a style, then choose [Bold On]Select.";
}
```

CaseOf 195:

```
tbox={ "► Highlight an outline style, then choose [Bold On]Select.";
}
```

CaseOf 240:

```
tbox={ "► Type the first heading for your outline, then press [Bold On]Enter[Bold Off] to move to";
      " the next line.";
      """;
}
```

CaseOf 250:

```
tbox={ "To change the heading level, press [Bold On]Tab[Bold Off] to move it in or [Bold On]Shift+Tab[Bold Off] to
move";
      "it out.";
      """;
      "► Adjust the heading level,";
      " then type your heading text,";
      " then press [Bold On]Enter[Bold Off].";
      """;
      "Repeat the above steps until you have created all the headings you need.";
      """;
}
```

CaseOf 260:

```
tbox={ "You have deleted the outline number. There are";
      "two ways to restore the outline number:";
      """;
      "Press [Bold On]Ctrl+T[Bold Off].";
      "-or-";
      "Place the cursor at the end of the heading above the";
      "line where the number is missing, and press [Bold On]Enter[Bold Off].";
}
```

CaseOf 310:

```
tbox={ "Before you end your outline,";  
      "";  
      "► Place the cursor at the end of your outline.";  
      "";  
    }
```

CaseOf 320:

```
tbox={ "You have now created a standard outline.";  
      "";  
      "The next activity in producing an outline";  
      "is to modify it—that is, add new headings,";  
      "delete headings, and so forth.";  
      "";  
      "Would you like a Coach to help you modify";  
      "your outline?";  
      "";  
    }
```

CaseOf 325:

```
tbox={ "You have successfully modified your outline.";  
      "";  
      "You can also change the outline to a";  
      "different style.";  
      "";  
      "Would you like a Coach to help you change";  
      "your outline to a different style?";  
      "";  
    }
```

CaseOf 400:

```
tbox={ "This Coach helps you apply a different outline";  
      "style to a previously created outline.";  
    }
```

CaseOf 410:

```
tbox={ "Before changing your outline style,";  
      "";  
      "► Place the cursor anywhere in the middle of your outline.";  
      "";  
    }
```

CaseOf 420:

```
tbox={ "To change your outline style,";  
      "";  
      "► Choose [Bold On]Tools[Bold Off],";
```

```
    " then [Bold On]Outline[Bold Off],";
    " then [Bold On]Outline Style[Bold Off].";
    " -or-";
    "▶ Press [Bold On]Ctrl+F5[Bold Off].";
    """;
}
```

CaseOf 500:

```
tbox={ "▶ Choose [Bold On]Begin New Outline.";
      }
```

CaseOf 510:

```
tbox={ "▶ Choose [Bold On]End Outline.";
      }
```

CaseOf 515:

```
tbox={ "▶ Choose [Bold On]Outline Style.";
      }
```

CaseOf 520:

```
tbox={ "You chose Outline Style. You would use this option";
      "to change the style of an outline that you had";
      "already created and had on you screen.";
      "▶ Choose [Bold On]Close[Bold Off] to exit this dialog and start over.";
      }
```

CaseOf 600:

```
tbox={ "This Coach helps you modify a previously created";
      "outline in two ways.";
      """;
      }
```

CaseOf 610:

```
tbox={ "Insert a new heading.";
      """;
      "▶ Place the cursor at the end of the heading above where you want to";
      " insert a new heading, ";
      " then press [Bold On]Enter[Bold Off], ";
      " then use [Bold On]Tab[Bold Off] and [Bold On]Shift+Tab[Bold Off] to change the heading level, ";
      " then type your heading. ";
      """;
      "Repeat the above steps until you have added all the headings you want.";
      """;
      }
```

CaseOf 620:

```
tbox={ "Change a heading level:";
      """,
      "► Place the cursor at the start of the heading you want to change,";
      " then use [Bold On]Tab[Bold Off] and [Bold On]Shift+Tab[Bold Off] to change the heading level.";
      """,
      "Repeat the above steps until you have changed all the headings you want.";
      """,
      }
```

CaseOf 800:

```
tbox={ "Coach is now finished. Would you";
      "like to return to its main menu?";
      """,
      }
```

CaseOf 1000:

```
tbox={ "No Outline has been selected.";
      """,
      "Would you like to try again?";
      """,
      }
```

CaseOf 1010:

```
tbox={ "No steps have been selected.";
      """,
      "Would you like to quit Coach?";
      """,
      }
```

CaseOf 1020:

```
tbox={ "The ""Headings"" style has not";
      "been selected. The style that";
      "was selected is called "" + CurSty + "".";
      """,
      "Would you like to try again?";
      """,
      }
```

CaseOf 1025:

```
tbox={ "A new outline style has not";
      "been selected.";
      """,
      "Would you like to try again?";
      """,
      }
```


CaseOf 2000:

```
tbox={ "The ""Outline"" style provides the standard outline style";  
      "such as the one shown below:";  
      "";  
      "I. Heading level one";  
      "  A. Heading level two";  
      "  B. Heading level two";  
      "    1. Heading level three";  
      "    2. Heading level three";  
      }
```

CaseOf 2010:

```
tbox={ "You must have an outline in your document";  
      "before running this Coach.";  
      "";  
      "Do you want to run Coach on Creating a";  
      "standard outline?";  
      "";  
      }
```

EndSwitch
ENDPROC