```
Private Sub cmbCloudEl_Click()
 ' MsgBox "hello world"
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "ACTIVE LEVEL ""Backcircle"""

    CadInputQueue.SendCommand "ACTIVE WEIGHT 3"
'   Set a variable associated with a dialog box
    SetCExpressionValue "tcb->symbology.color", 1, "MGDSHOOK"


'   Start a command
    CadInputQueue.SendCommand "PLACE REVCLOUD ELEMENT"

'    SetCExpressionValue "cloudParams.radius", (ActiveModelReference.UORsPerMasterUnit * 0.1), "COMPCURV"

End Sub

Private Sub cmbCloudPt_Click()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "ACTIVE LEVEL ""Backcircle"""

    CadInputQueue.SendCommand "ACTIVE WEIGHT 3"

'   Set a variable associated with a dialog box
    SetCExpressionValue "tcb->symbology.color", 1, "MGDSHOOK"

'    SetCExpressionValue "cloudParams.radius", (ActiveModelReference.UORsPerMasterUnit * 0.1), "COMPCURV"


'   Start a command
    CadInputQueue.SendCommand "PLACE REVCLOUD POINTS"

End Sub

Private Sub cmbComprLevels_Click()
Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "MDL KEYIN COMPRESSOPTIONS DIALOG COMPRESSOPTIONS"

    CadInputQueue.SendCommand "COMPRESS OPTIONS COMPRESS"

'   Remove the following line to let the user close the dialog box.
    DialogResult = msdDialogBoxResultOK

    'End If  ' Compress Options

    CadInputQueue.SendCommand "ACTIVE LEVEL ""Level 1"""

    CadInputQueue.SendCommand "MDL KEYIN lvlmangr levelmanager dialog open"

'   The following statement opens modal dialog "Level/Filter Import"

    Dim modalHandler As New Macro1ModalHandler2
    AddModalDialogEventsHandler modalHandler

'   The following statement opens modal dialog "Import Levels"
```

```
    CadInputQueue.SendCommand "LEVELMANAGER LIBRARY IMPORT"

    RemoveModalDialogEventsHandler modalHandler
    CommandState.StartDefaultCommand
End Sub

Private Sub cmbDataFieldsOffView5_Click()
  Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Coordinates are in master units
    startPoint.x = -2.38719521710856
    startPoint.Y = 5.40902255639098
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 5

'   Send a keyin that can be a command string
    CadInputQueue.SendKeyin "dialog viewsettings popup"

    CadInputQueue.SendKeyin "MDL KEYIN BENTLEY.VIEWATTRIBUTESDIALOG,VAD VIEWATTRIBUTESDIALOG SETATTRIBUTE 4 DataFields False"

    CommandState.StartDefaultCommand

End Sub

Private Sub cmbExistingLevel_Click()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "ACTIVE LEVEL ""Existing"""

'   Set a variable associated with a dialog box
    SetCExpressionValue "tcb->symbology.color", 0, "MGDSHOOK"

    CadInputQueue.SendCommand "ACTIVE STYLE 0"

    CadInputQueue.SendCommand "ACTIVE WEIGHT 1"

    CommandState.StartDefaultCommand
End Sub

Private Sub cmbDFence_Click()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "INPUTMANAGER MENU -705,2"

    CadInputQueue.SendCommand "PLACE FENCE ICON"

'   Send a tentative point
'   Coordinates are in master units
    CadInputQueue.SendTentativePoint Point3dFromXYZ(0.023591, 21.978293, 0#), 1

'   Coordinates are in master units
    startPoint.x = 0#
    startPoint.Y = 22#
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

'   Set a variable associated with a dialog box
    SetCExpressionValue "tcb->msToolSettings.fence.placeMode", 0, ""

    CadInputQueue.SendCommand "PLACE FENCE ICON"

    CadInputQueue.SendTentativePoint Point3dFromXYZ(0.005714, 21.91996, 0#), 1

    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(33.960146, 0.004824, 0#), 1

    point.x = startPoint.x + 34#
    point.Y = startPoint.Y - 22#
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "WINDOW AREA EXTENDED 1"

    point.x = startPoint.x + 32.745686
    point.Y = startPoint.Y - 12.483273
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 32.803723
    point.Y = startPoint.Y - 22.152551
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "FIT VIEW EXTENDED 5"

    CadInputQueue.SendCommand "DIALOG PLOT"

    '   Set a variable associated with a dialog box
    SetCExpressionValue "plotUI.uiPlotArea", 2, "PLOTDLG"

    '   Set a variable associated with a dialog box TO MONOCHROME
    SetCExpressionValue "plotUI.uiColorMode", 0, "PLOTDLG"

    ' Set raster quality to 100%



    'Done with raster quality set to 100%

    'MACRO 6 TURNS LINE WEIGHTS OFF
    'MsgBox ("MACRO 6 NEXT")
    Macro8

    'CadInputQueue.SendCommand "PRINT MAXIMIZE"

    CadInputQueue.SendKeyin "LOCK SNAP KEYpoint"

    CadInputQueue.SendCommand "LOCK SNAP ON"

    CadInputQueue.SendCommand "LOCK UNIT ON"

    CadInputQueue.SendCommand "LOCK AXIS OFF"

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    CommandState.StartDefaultCommand

End Sub

Private Sub cmdBlueRev1_Click()
```

```
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Coordinates are in master units
    startPoint.x = 32.5918877528005
    startPoint.Y = 12.4385990697246
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

'   Send a message string to an application
'   Content is defined by the application
    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 17"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 7"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 17"

    point.x = startPoint.x - 1.30742656678038E-02
    point.Y = startPoint.Y - 2.26741425634707E-02
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 9.58779482305516E-03
    point.Y = startPoint.Y + 2.52903897823344E-02
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 8.71617711190709E-04
    point.Y = startPoint.Y + 8.7208240628911E-04
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 6.97294168949014E-03
    point.Y = startPoint.Y - 4.36041203143667E-03
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 2.09188250684775E-02
    point.Y = startPoint.Y - 6.10457684400956E-03
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 1.74323542237289E-02
    point.Y = startPoint.Y - 2.18020601571833E-02
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 2.26620604908589E-02
    point.Y = startPoint.Y - 2.61624721886182E-02
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 4.61957386929015E-02
    point.Y = startPoint.Y - 2.18020601571833E-02
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 2.70201490467983E-02
    point.Y = startPoint.Y + 4.36041203143684E-02
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 1.22026479566202E-02
    point.Y = startPoint.Y + 5.75574388149658E-02
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 4.35808855593933E-03
    point.Y = startPoint.Y + 7.93594989721509E-02
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 1.74323542237431E-03
    point.Y = startPoint.Y + 4.36041203143667E-03
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 2.26620604908518E-02
    point.Y = startPoint.Y - 2.00578953446087E-02
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 16"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 7"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 16"

    point.x = startPoint.x + 5.66551512271403E-02
    point.Y = startPoint.Y + 8.63361582224496E-02
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 6.12311442108719E-02
    point.Y = startPoint.Y + 0.22020080758756
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 27"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 7"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 27"

    point.x = startPoint.x - 9.72058033375589E-03
    point.Y = startPoint.Y + 1.05078094842259
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 83"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 7"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 83"

    point.x = startPoint.x + 0.118407223210703
    point.Y = startPoint.Y + 1.18966007162385
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 0.026733830289345
    point.Y = startPoint.Y + 7.14537627062998
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 7"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    point.x = startPoint.x + 4.14151411121395E-02
    point.Y = startPoint.Y + 7.54758340166723
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1
    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 7"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    point.x = startPoint.x + 6.88710990145225E-02
    point.Y = startPoint.Y + 7.50942979639216
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 2.72785913588223E-02
    point.Y = startPoint.Y + 7.94079718288965
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 7"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    point.x = startPoint.x + 8.74202134307112E-02
    point.Y = startPoint.Y + 8.00097086892347
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 4.08431419891642E-02
    point.Y = startPoint.Y + 8.33546897688507
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 7"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    point.x = startPoint.x + 0.110572558884108
    point.Y = startPoint.Y + 8.31715524635304
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 5.35360749083154E-02
    point.Y = startPoint.Y + 8.74431210998267
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 7"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    point.x = startPoint.x + 7.00968114208607E-02
    point.Y = startPoint.Y + 8.69983590726201
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CommandState.StartDefaultCommand



End Sub

Private Sub cmdEsizeFence_Click()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long
```

```
'   Start a command
    CadInputQueue.SendCommand "PLACE FENCE ICON"

'   Send a tentative point
'   Coordinates are in master units
    CadInputQueue.SendTentativePoint Point3dFromXYZ(0.493826490298015, 30.0246028833885, 0#), 1

'   Coordinates are in master units
    startPoint.x = 0#
    startPoint.Y = 30#
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(41.97131768619, 1.17293206833481E-02, 0#), 1

    point.x = startPoint.x + 42.0000000000001
    point.Y = startPoint.Y - 30#
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    point.x = startPoint.x + 1.79290742839691
    point.Y = startPoint.Y + 3.46071529917275
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 5

    CadInputQueue.SendCommand "PRINT MAXIMIZE"

    CadInputQueue.SendCommand "WINDOW AREA EXTENDED 1"

    point.x = startPoint.x + 40.9646372074583
    point.Y = startPoint.Y - 30.3439460111004
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 40.801902248909
    point.Y = startPoint.Y - 20.3922465283965
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CommandState.StartDefaultCommand
End Sub

Private Sub cmdFence18000_Click()
  Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "PLACE FENCE ICON"

'   Send a tentative point
'   Coordinates are in master units
    CadInputQueue.SendTentativePoint Point3dFromXYZ(0.098639241090924, 14.4992497162524, 1.4111111111138), 1

'   Coordinates are in master units
    startPoint.x = 0#
    startPoint.Y = 14.6666666666666
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(22.6630469909838, -4.14684863753751E-03, 1.41111111111402), 1
    point.x = startPoint.x + 22.6666666666667
    point.Y = startPoint.Y - 14.6666666666666
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    point.x = startPoint.x + 23.1448617967697
    point.Y = startPoint.Y - 6.5622857142857
    point.Z = startPoint.Z + 1.4111111111123
    CadInputQueue.SendDataPoint point, 5

    CadInputQueue.SendCommand "FIT VIEW EXTENDED 5"

    CadInputQueue.SendCommand "WINDOW AREA EXTENDED 1"

    point.x = startPoint.x + 21.7245776434224
    point.Y = startPoint.Y - 8.34410349586891
    point.Z = startPoint.Z + 1.41111111111421
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 21.6370483654411
    point.Y = startPoint.Y - 14.7128887948723
    point.Z = startPoint.Z + 1.41111111111426
    CadInputQueue.SendDataPoint point, 1

    CommandState.StartDefaultCommand

End Sub

Private Sub cmdFence6000_Click()

    CommandState.StartDefaultCommand

    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "PLACE FENCE ICON"

'   Send a tentative point
'   Coordinates are in master units
    CadInputQueue.SendTentativePoint Point3dFromXYZ(0.365598749417673, 43.5815800805484, 4.23333333333436), 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(0.102272749327261, 44.0333269768638, 4.23333333333436), 1

'   Coordinates are in master units
    startPoint.x = 0#
    startPoint.Y = 43.9999999999999
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(67.9812786286272, 3.29940720802703E-02, 4.23333333333444), 1

    point.x = startPoint.x + 68#
    point.Y = startPoint.Y - 43.9999999999999
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    point.x = startPoint.x + 8.83333333333333
    point.Y = startPoint.Y + 56.1666666666667
    point.Z = startPoint.Z + 4.23333333333333
    CadInputQueue.SendDataPoint point, 5

    CadInputQueue.SendCommand "FIT VIEW EXTENDED 5"
    CadInputQueue.SendCommand "WINDOW AREA EXTENDED 1"

    point.x = startPoint.x + 65.6749694500775
    point.Y = startPoint.Y - 24.7501961741185
    point.Z = startPoint.Z + 4.23333333333444
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 65.7208898602495
    point.Y = startPoint.Y - 44.117079717327
    point.Z = startPoint.Z + 4.23333333333444
    CadInputQueue.SendDataPoint point, 1

    CommandState.StartDefaultCommand

End Sub



Private Sub cmdFenceDonE_Click()
   Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "PLACE FENCE ICON"

'   Send a tentative point
'   Coordinates are in master units
    CadInputQueue.SendTentativePoint Point3dFromXYZ(-7.68330441137402, 29.5718401832303, 0#), 1

'   Coordinates are in master units
    startPoint.x = -8.00000000000005
    startPoint.Y = 30#
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(33.9605377145485, 2.06433146363044E-02, 0#), 1

    point.x = startPoint.x + 42#
    point.Y = startPoint.Y - 30#
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    point.x = startPoint.x - 4.59999999999996
    point.Y = startPoint.Y - 14.05
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 5

    CadInputQueue.SendCommand "FIT VIEW EXTENDED 5"

    CadInputQueue.SendCommand "WINDOW AREA EXTENDED 1"

    point.x = startPoint.x + 40.7100968867561
    point.Y = startPoint.Y - 20.425106943707
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 40.9609209462061
    point.Y = startPoint.Y - 30.0283192740974
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CommandState.StartDefaultCommand

End Sub
```

```
Private Sub cmdNewOrRevLevel_Click()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "ACTIVE LEVEL ""New or Revisions"""

    CadInputQueue.SendCommand "ACTIVE STYLE 0"

    CadInputQueue.SendCommand "ACTIVE WEIGHT 1"

'   Set a variable associated with a dialog box
    SetCExpressionValue "tcb->symbology.color", 7, "MGDSHOOK"

    CommandState.StartDefaultCommand
End Sub

Private Sub cmdSetRadius_Click()
 Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "PLACE REVCLOUD POINTS"

'   Set a variable associated with a dialog box
    SetCExpressionValue "cloudParams.flags.lockRadius", 1, "COMPCURV"

    SetCExpressionValue "cloudParams.radius", (ActiveModelReference.UORsPerMasterUnit * 0.1), "COMPCURV"

    CommandState.StartDefaultCommand
End Sub

Private Sub cmdTextDialog_Click()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

    Dim modalHandler As New Macro1ModalHandler5
    AddModalDialogEventsHandler modalHandler

'   The following statement opens modal dialog "Preferences [descartes]"

'   Start a command
    CadInputQueue.SendCommand "MDL SILENTLOAD USERPREF"

    CadInputQueue.SendCommand "MDL SILENTUNLOAD SPELLCHECK"

    RemoveModalDialogEventsHandler modalHandler
    CommandState.StartDefaultCommand
End Sub

Private Sub cmdTitleBlock_Click()
  Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command

    CadInputQueue.SendCommand "ACTIVE LEVEL ""Border-titleblock"""

'   Set a variable associated with a dialog box
   ' SetCExpressionValue "tcb->symbology.color", -1, "MGDSHOOK"

    'CadInputQueue.SendCommand "ACTIVE WEIGHT 0"

    CommandState.StartDefaultCommand
End Sub
```

```
Private Sub cmdWordProcessor_Click()
  Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

    Dim modalHandler As New Macro2ModalHandler1
    AddModalDialogEventsHandler modalHandler

'   The following statement opens modal dialog "Preferences [descartes]"

'   Start a command
    CadInputQueue.SendCommand "MDL SILENTLOAD USERPREF"

    CadInputQueue.SendCommand "MDL SILENTUNLOAD SPELLCHECK"

    RemoveModalDialogEventsHandler modalHandler
    CommandState.StartDefaultCommand
End Sub


Private Sub CommandButton1_Click()
 Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "ACTIVE LEVEL ""Notes and Reference"""

'   Set a variable associated with a dialog box
    SetCExpressionValue "tcb->symbology.color", 7, "MGDSHOOK"

    CommandState.StartDefaultCommand
End Sub



Private Sub UserForm_Click()

End Sub
```

```
Private Sub cmdFolderPath_Click()
    'Get the folder of drawings and insert the path into the form
    lblFolderPath.Caption = modNextFile.SelectDGNFolder

    If lblFolderPath.Caption = "" Then
      MsgBox "You pressed Cancel, or you did not select a file inside of the folder."
      Exit Sub
    End If

    'save path to desktop\Filelists\path.txt
    modNextFile.WritePathToFile FullFolderPath:=lblFolderPath.Caption

    'create a text file of all of the DGN files in the folder
    Call modNextFile.FileCreate(lblFolderPath)

    'Create an array from the text file and insert the drawing names into
    'the listbox
    Call modNextFile.InsertFileLinesToArray(modNextFile.GetFileListPath)

End Sub

Private Sub cmdLoadPrevious_Click()
    Dim path As String
   'get path from path.txt file
    path = GetPathToFile

   'put path on label
   lblFolderPath.Caption = path

    'create a text file of all of the DGN files in the folder
    Call modNextFile.FileCreate(path)

    'Create an array from the text file and insert the drawing names into
    'the listbox
    Call modNextFile.InsertFileLinesToArray(modNextFile.GetFileListPath)

End Sub

Private Sub ListBox1_Click()

    Dim path As String
    Dim filename As String
    Dim ReadOnly As Boolean

    path = lblFolderPath.Caption
    'filename = ListBox1.Selected(pvargindex)
    filename = ListBox1.Value
    'MsgBox filename
    filename = path & filename

    If togReadOnly.Caption = "Read Only" Then
      ReadOnly = True
     Else
      ReadOnly = False
    End If

    modNextFile.OpenFile filename:=filename, EditMode:=ReadOnly

    'more code to make read only open with extents in view 1

     If ReadOnly Then
       'MsgBox "This is read only mode"
     End If
End Sub

Private Sub ListBox1_KeyPress(ByVal KeyAscii As MSForms.ReturnInteger)
  MsgBox "a key was pressed"
End Sub

Private Sub togReadOnly_Click()
```

```
    If togReadOnly.Caption = "Read Only" Then
      togReadOnly.Caption = "Edit Mode"
     Else
      togReadOnly.Caption = "Read Only"
   End If
End Sub

Private Sub UserForm_Initialize()
  togReadOnly.Caption = "Read Only"
End Sub


Private Sub UserForm_KeyPress(ByVal KeyAscii As MSForms.ReturnInteger)
  Select Case KeyAscii
    Case vbKeyF2
      MsgBox "f2 pressed"
    Case Else
      KeyAscii = 0
    End Select
End Sub
```

```
Private Sub CommandButton1_Click()

End Sub

Private Sub CommandButton1_KeyDown(ByVal KeyCode As MSForms.ReturnInteger, ByVal Shift As Integer)
  Call custom_KeyDown(KeyCode, Shift)
End Sub

Private Sub CommandButton2_Click()

End Sub

Private Sub CommandButton2_KeyDown(ByVal KeyCode As MSForms.ReturnInteger, ByVal Shift As Integer)
  Call custom_KeyDown(KeyCode, Shift)
End Sub

Private Sub custom_KeyDown(ByVal KeyCode As MSForms.ReturnInteger, ByVal Shift As Integer)
  Dim Icount As Integer
  Dim TooHigh As Integer
  Const TooLow = 0
  Const No_Selection = -1
  Const EnterKey = 13

  Select Case KeyCode
     Case EnterKey
       Label1.Caption = "Pressed ENTER"
     Case vbKeyF2
       Label1.Caption = "F2 pressed"
     Case vbKeyUp
       Label1.Caption = "Up arrow"
       Icount = ListBox1.ListIndex
       TooHigh = ListBox1.ListCount - 1

       'MsgBox str(icount)
        Select Case Icount
          Case No_Selection
            'Select first item
            ListBox1.ListIndex = 0
          Case TooLow
            'Wrap back to first item
            ListBox1.ListIndex = ListBox1.ListCount - 1
          Case Else
            ListBox1.ListIndex = Icount - 1
        End Select
    Case vbKeyDown
     Label1.Caption = "down arrow"
     Icount = ListBox1.ListIndex
     TooHigh = ListBox1.ListCount - 1
     'MsgBox str(icount)
        Select Case Icount
          Case No_Selection
            'Select first item
            ListBox1.ListIndex = 0
          Case TooHigh
            'Wrap back to first item
            ListBox1.ListIndex = 0
          Case Else
            ListBox1.ListIndex = Icount + 1
        End Select
   Case Else
    'do nothing
  End Select
End Sub

Private Sub ListBox1_Click()
  'Label1.Caption = "you clicked" & " " & ListBox1.Value & "listed at:" & str(ListBox1.ListIndex)
End Sub

Private Sub ListBox1_KeyDown(ByVal KeyCode As MSForms.ReturnInteger, ByVal Shift As Integer)
  Call custom_KeyDown(KeyCode, Shift)
End Sub
```

```
Private Sub UserForm_Click()

End Sub

Private Sub UserForm_Initialize()
  ListBox1.AddItem "first"
  ListBox1.AddItem "second"
  ListBox1.AddItem "third"
  ListBox1.AddItem "fourth"
  ListBox1.AddItem "fifth"
  ListBox1.AddItem "sixth"
  ListBox1.AddItem "seventh"

End Sub
```

```
Private Sub ListView1_BeforeLabelEdit(Cancel As Integer)

End Sub

Private Sub TextBox1_KeyUp(ByVal KeyCode As MSForms.ReturnInteger, ByVal Shift As Integer)
    Dim v2 As String

    TextBox2.Text = TextBox1.CurLine
    TextBox3.Text = TextBox1.CurX
    TextBox4.Text = TextBox1.CurTargetX
    TextBox5.Text = TextBox1.TabKeyBehavior

    v2 = KeyCode


    MsgBox v2




End Sub
Private Sub UserForm_Initialize()
    TextBox1.MultiLine = True

    TextBox1.Text = "Type your text here. User CTRL + ENTER to start a new line."
End Sub
```

```
Option Explicit
Option Base 1

Sub testUserForm1()
  UserForm1.show vbModeless
End Sub

Sub test_SeeAttachment()
 Dim message As String
 message = SeeAttachment
 MsgBox message
End Sub

Function SeeAttachment() As String

  Dim strRtrnMsg As String
  Dim RasterFullName As String
  Dim RasterPath As String
  Dim DesignFileName As String
  Dim DesignFilePath As String
  Dim Icount As Integer
  Dim intAttachCount As Integer

  Dim strAttachName As String
  Dim strAttachPath As String

  Dim strFilename As String
  Dim strFilepath As String

  Dim currentFile As DesignFile
  Set currentFile = Application.ActiveDesignFile

  strFilename = currentFile.Name
  strFilepath = currentFile.path

  Dim RasterCount As Integer
        'gets the name of the first attached raster and the number of rasters attached
        'need full path since raster could be in another folder
   Dim att As Rasters
   Set att = Application.RasterManager.Rasters
        'are there any attachments
   intAttachCount = att.count
   strAttachPath = att.Item(1).RasterInformation.path
   strAttachName = att.Item(1).RasterInformation.FullName


   If intAttachCount = 0 Then
         SeeAttachment = ""
        Exit Function
   End If

  strRtrnMsg = ""  'keep track of all issues
  'Test Path Name
        If StrComp(strAttachPath, strFilepath, vbTextCompare) = 0 Then
            'path match
          Else
            strRtrnMsg = "Raster Path Different! " & vbLf
        End If

  'Test File Name
        'take of .cit and .dgn filename
        strAttachName = Left(strAttachName, Len(strAttachName) - 4)
        strFilename = Left(strFilename, Len(strFilename) - 4)
        If StrComp(strAttachName, strFilename, vbTextCompare) = 0 Then
             'name match
          Else
             strRtrnMsg = strRtrnMsg & "File and Raster Names are Different! " & vbLf
        End If

  'Test for too many rasters attached
        If intAttachCount >= 2 Then
        strRtrnMsg = strRtrnMsg & "Too many Rasters!"
        End If

   SeeAttachment = strRtrnMsg

End Function

Sub SeeFileName()
  Dim fna As DesignFile
  Set fna = Application.ActiveDesignFile
  MsgBox fna.Name
  MsgBox fna.path
End Sub

Sub FileAttributes()
  Dim message As String
  Dim SnapE As Boolean
  Dim UnitL As Boolean
  Dim graphG As Boolean
  Dim activeR As Boolean
  Dim ActRefMod As ModelReference

  With Application.ActiveSettings
      .SnapLockEnabled = True
      .UnitLockEnabled = True
      .GraphicGroupLockEnabled = True
        '    .GridUnits
        '    .GridReference
       .AxisLockEnabled = False
       .GridLockEnabled = False

  End With

  CadInputQueue.SendKeyin "LOCK SNAP KEYpoint"

 If activeR = Application.HasActiveModelReference Then
   Set ActRefMod = Application.ActiveModelReference
 End If

   MsgBox SnapE & UnitL

End Sub


Sub F2_G01()  'G --> retive information and reset settings before Getting Out Of Drawing
    'by Keith Knowles 12/10/2013

    Dim message As String
    Dim bdrElement As CellElement
    Set bdrElement = GetBorder(False)
    'MsgBox bdrElement.Name & "made it"

    'if no border then skip over GetThe Points and instead use
    'another routine, that does extents and puts a fence about extents

    'put on the fence
    message = GetThePoints(bdrElement)
    message = TableColor_G01.ColorTable

    If ActiveDesignFile.Views(5).IsOpen Then
        CadInputQueue.SendCommand "FIT VIEW EXTENDED 5"
      Else
        message = message & "FIX VIEW 5!"
    End If

     If message <> "" Then
       ShowStatus message
     End If

End Sub
Function GetThePoints(BDR As CellElement) As String
```

```
     Dim delta_Y As Variant
     Dim delta_X As Variant
     Dim D12_Ratio As Double
     D12_Ratio = 1.54545454545455
     Const E12_Ratio As Double = 1.4
     Dim FortyTwo As Variant
     FortyTwo = 42#
     Dim ThirtyFour As Variant
     ThirtyFour = 34#
     Dim curElem As Element

     Dim pts(1 To 4) As Point3d
        'BDR.Origin.x

        'If non-scaled drawings
      If BDR.Name = "BDR-D10" Then
        pts(1).x = BDR.Range.Low.x
        pts(1).Y = BDR.Range.Low.Y
        pts(2).x = BDR.Range.Low.x + ThirtyFour
        pts(2).Y = BDR.Range.Low.Y
        pts(3).x = BDR.Range.Low.x + ThirtyFour
        pts(3).Y = BDR.Range.High.Y
        pts(4).x = BDR.Range.Low.x
        pts(4).Y = BDR.Range.High.Y

       ElseIf BDR.Name = "BDR-E10" Then
        pts(1).x = BDR.Range.Low.x
        pts(1).Y = BDR.Range.Low.Y
        pts(2).x = BDR.Range.Low.x + FortyTwo
        pts(2).Y = BDR.Range.Low.Y
        pts(3).x = BDR.Range.Low.x + FortyTwo
        pts(3).Y = BDR.Range.High.Y
        pts(4).x = BDR.Range.Low.x
        pts(4).Y = BDR.Range.High.Y


       ElseIf BDR.Name = "BDR-T10" Or BDR.Name = "BDR-T12" Then
        pts(1).x = BDR.Range.Low.x
        pts(1).Y = BDR.Range.Low.Y
        pts(2).x = BDR.Range.High.x
        pts(2).Y = BDR.Range.Low.Y
        pts(3).x = BDR.Range.High.x
        pts(3).Y = BDR.Range.High.Y
        pts(4).x = BDR.Range.Low.x
        pts(4).Y = BDR.Range.High.Y

       ElseIf BDR.Name = "BDR-D12" Then
        delta_Y = BDR.Range.High.Y - BDR.Range.Low.Y
        'deduce delta_X by  Ratio 17/11
        delta_X = D12_Ratio * delta_Y
        pts(1).x = BDR.Range.Low.x
        pts(1).Y = BDR.Range.Low.Y
        pts(2).x = BDR.Range.Low.x + delta_X
        pts(2).Y = BDR.Range.Low.Y
        pts(3).x = BDR.Range.Low.x + delta_X
        pts(3).Y = BDR.Range.High.Y
        pts(4).x = BDR.Range.Low.x
        pts(4).Y = BDR.Range.High.Y

       ElseIf BDR.Name = "BDR-E12" Then
        delta_Y = BDR.Range.High.Y - BDR.Range.Low.Y
        'deduce delta_X by ratio 14/10
        delta_X = E12_Ratio * delta_Y
        pts(1).x = BDR.Range.Low.x
        pts(1).Y = BDR.Range.Low.Y
        pts(2).x = BDR.Range.Low.x + delta_X
        pts(2).Y = BDR.Range.Low.Y
        pts(3).x = BDR.Range.Low.x + delta_X
        pts(3).Y = BDR.Range.High.Y
        pts(4).x = BDR.Range.Low.x
        pts(4).Y = BDR.Range.High.Y
       Else
         MsgBox "No border on drawing!"
       End If

    ' if fence happens
    ' GetThePoints = True
    With ActiveDesignFile.Fence
       .DefineFromModelPoints 1, pts()
       .Draw msdDrawingModeHilite
    End With


    If BDR.Range.Low.x <> 0 Or BDR.Range.Low.Y <> 0 Then
      GetThePoints = "BORDER OFF 0,0! "
    End If

End Function
Function GetBorder(ignoreT As Boolean) As Element

  Dim number As Double


  Dim rngBDR As Range3d
  Dim pntBDRs As Point3d
  Dim pntBDRe As Point3d
  Dim rngTBDR As Range3d
  Dim pntTBDRs As Point3d
  Dim pntTBDRe As Point3d
  Dim dblScale As Double

  Dim BorderName As String
  Dim oElem As Element
  Dim oCellElem As CellElement
  Dim BdrObject As CellElement
  Dim TbdrObject As CellElement
  Dim oEnum As ElementEnumerator
  Dim ElementCounter As Long
  Dim BorderType As String

  Dim BorderD10 As Boolean
  Dim BorderE10 As Boolean
  Dim BorderT10 As Boolean

  Dim BorderD12 As Boolean
  Dim BorderE12 As Boolean
  Dim BorderT12 As Boolean

  BorderD10 = False
  BorderE10 = False
  BorderT10 = False
  BorderD12 = False
  BorderE12 = False
  BorderT12 = False

  Set oEnum = ActiveModelReference.Scan()

  While oEnum.MoveNext
        ElementCounter = ElementCounter + 1
        Set oElem = oEnum.Current

        If oElem.IsCellElement Then
          Set oCellElem = oElem

          Select Case oCellElem.Name
            Case "BDR-D10"
              'MsgBox "D10"
              Set BdrObject = oCellElem
              BorderD10 = True
            Case "BDR-D12"
              'MsgBox "D12"
              Set BdrObject = oCellElem
              BorderD12 = True
            Case "BDR-E10"
              'MsgBox "E10"
              Set BdrObject = oCellElem
              BorderE10 = True
            Case "BDR-E12"
              'MsgBox "E12"
              Set BdrObject = oCellElem
              BorderE12 = True
            Case "BDR-T10"
              'MsgBox "T10"
              Set TbdrObject = oCellElem
              BorderT10 = True
            Case "BDR-T12"
              'MsgBox "T12"
              Set TbdrObject = oCellElem
              BorderT12 = True
            Case Else
              'do nothing
           End Select
        End If
   Wend

   If ignoreT = True Then
     'MsgBox "ignore T"
       If BorderE10 Or BorderD10 Or BorderD12 Or BorderE12 Then

          Set GetBorder = BdrObject
          Exit Function
         Else
          MsgBox "No D or E Borders in this file"
       End If
   End If

   If ignoreT = False Then
       If BorderT10 Or BorderT12 Then
          Set GetBorder = TbdrObject
          'MsgBox "T border takes priority"
         ElseIf BorderE10 Or BorderD10 Or BorderD12 Or BorderE12 Then
          Set GetBorder = BdrObject
          Exit Function
         Else
          MsgBox "No D or E Borders in this file"
       End If
   End If

End Function
```

```
Option Base 1
Option Explicit

Sub test_FoundAllLevels()
  Dim GotAllLevels As Boolean
  GotAllLevels = FoundAllLevels

    If GotAllLevels = False Then
       MsgBox "LEVELS"  'LEVELS PROBLEM
      Else
       MsgBox ""  'ALL LEVELS IN SUBSTATION.LEVELS WERE FOUND, SO RETURN NOTHING
    End If

End Sub
Function FoundAllLevels() As Boolean  'return True if all levels found

  Dim myLevel As Level
  Dim LevelCounter As Integer
  Dim ICountLevels As Integer
  Dim Icount As Integer
  Dim ICountChange As Boolean


  Dim strLevel(23) As String
  strLevel(1) = "Text"
  strLevel(2) = "Property line"
  strLevel(3) = "Backcircle"
  strLevel(4) = "Border-titleblock"
  strLevel(5) = "DIMENSIONS"
  strLevel(6) = "New or Revisions"
  strLevel(7) = "Baselines"
  strLevel(8) = "Fence"
  strLevel(9) = "Removal or Abandoned"
  strLevel(10) = "Contours 1 ft"
  strLevel(11) = "Contours 5 ft"
  strLevel(12) = "Liner Seal to Concrete"
  strLevel(13) = "Liner Extent"
  strLevel(14) = "Notes and References"
  strLevel(15) = "Material Item"
  strLevel(16) = "Vendor"
  strLevel(17) = "Design Master(Red)"
  strLevel(18) = "Existing"
  strLevel(19) = "Mark List"
  strLevel(20) = "Default"
  strLevel(21) = "Fence Corners"
  strLevel(22) = "Centerlines"
  strLevel(23) = "Foundations"

  Icount = 1
  ICountLevels = 0
  'ICountChange = False


  For Each myLevel In ActiveDesignFile.Levels

        For Icount = 1 To UBound(strLevel)
          If myLevel.Name = strLevel(Icount) Then
             Debug.Print myLevel.Name
             ICountLevels = ICountLevels + 1
             'ICountChange = True
             Exit For
          Else
             'do nothing
             'ICountChange = False
          End If
        Next Icount

       'count up unused levels
                'test to see if ICountLevels remained the same, if so then
                'the level tested wasn't a Good level, so try to delete it
                'Test to see if level might still be in use
                'If ICountChange = False Then

                   ' If myLevel.IsInUse Then
                          'do nothing, because level is being used
                       ' Else        'delete level
                           '*****the following code gave error "Level Id is invalid"
                           ' ActiveDesignFile.DeleteLevel myLevel

                           'ActiveModelReference.MasterUnit.System
                           'activemodelreference.

                           'ActiveModelReference.Levels.

                           ' ActiveDesignFile.Levels.Rewrite
                    ' End If
                 'End If
  Next

  If ICountLevels = UBound(strLevel) Then
    FoundAllLevels = True
  Else
    FoundAllLevels = False
  End If

End Function
```

```
Option Base 1
Declare Function mdlDialog_fileOpen Lib _
"stdmdlbltin.dll" (ByVal _
 filename As String, ByVal rFileH As Long, ByVal _
 resourceId As Long, ByVal suggestedFileName As String, _
 ByVal filterString As String, _
 ByVal defaultDirectory As String, _
 ByVal titleString As String) As Long
Sub NextFile()
  frmNextFile.show vbModeless
End Sub
Sub TestInsertFileLinesToArray()
   Dim fpath As String
   fpath = GetFileListPath
   InsertFileLinesToArray (fpath)
End Sub
Sub InsertFileLinesToArray(filepath As String)
  'this still needs to be updated
  'the routine is to be used to load an array and then to load a list box
  'will use a two-dimensional array so that a number is associated with each
  'file name, so that the file can be opened and the current file can be saved to
  'a text file ... this has yet to be coded.

  Dim myArray() As String    ' Declare dynamic array.
  Dim FileToOpen As String
  Dim I As Integer
  Dim arraySize As Integer
  Dim x As Integer
  Dim a As Integer
  Dim batchfile As String

  'clear ListBox1 before adding items to it
  frmNextFile.ListBox1.Clear

  Dim FFile As Long
  FFile = FreeFile
  'BatchFile = "C:\filelist.txt"
  batchfile = filepath
  Open batchfile For Input As #FFile
  I = 1
  While EOF(FFile) = False
     Line Input #FFile, FileToOpen
     'MsgBox FileToOpen
     'insert the files to an array
        ReDim Preserve myArray(I)      ' Re-allocate
        myArray(I) = FileToOpen    ' Initialize array.
        'MsgBox "array has:" & myArray(i)
        'frmNextFile.ListBox1.AddItem (myArray(i))
        I = I + 1
  Wend

     'loads all files
     frmNextFile.ListBox1.List() = myArray


    ' x = UBound(myArray)
     'a = LBound(myArray)
     'MsgBox Str(x) & " " & Str(a)
     'close the file
     Close FFile
End Sub



Function OpenFile(filename As String, EditMode As Boolean) As Boolean
  Application.OpenDesignFile filename, EditMode
End Function

Sub TestPickAFolder()
  Dim Folder As String
  Dim path As String
  Folder = PickAFolder
  MsgBox RootFolder
End Sub

Sub TestDesktopPathFunction()
   Dim strLine As String
   Dim strPath As String
   strPath = DesktopPath
   MsgBox strPath
End Sub


Function DesktopPath() As String
   Dim objFolders As Object
   Set objFolders = CreateObject("wScript.Shell").specialfolders
   DesktopPath = objFolders("desktop")
End Function


Function fileNamesInTextFile()
    Dim filepath As String
    Dim filename As String
    Dim First As Boolean
    Dim count As Integer

    Dim Folderpath As String
    Dim myFSO As New Scripting.FileSystemObject
    Dim myFolder As Scripting.Folder
    Dim myFile As Scripting.File
    Dim RootFolder As String
    'RootFolder = InputBox("Enter Root Folder:")
    RootFolder = PickAFolder(Folderpath)

    Set myFolder = myFSO.GetFolder(RootFolder)
    First = True
    count = 1
    For Each myFile In myFolder.Files
       Select Case UCase(Right(myFile.Name, 3))
         Case "DGN"
           If First = True Then
             Open filepath For Output As #1
               Print #1, Str(count) & myFile.path
               First = False
             Close #1
           Else
             Open filepath For Append As #1
               Print #1, Str(count) & " " & Right(myFile.path, 14)
             Close #1
           Else
             Open filepath For Append As #1
               Print #1, Str(count) & " " & Right(myFile.path, 14)  'myFile.Path
             Close #1
           End If
           count = count + 1
    End Select
    Next

End Function

Sub SelectDTopFile()

  Dim fname As String
  Dim strPath As String
  Dim FilesFolder As String
  Dim objFolders As Object
  Set objFolders = CreateObject("wScript.Shell").specialfolders

  DTopPath = objFolders("desktop")

  strPath = DTopPath & "\"      'uses function in this module, string path could be hard coded
  strPath = strPath & "Filelists"
  If (Dir(strPath, vbDirectory) <> "") Then
     'do nothing
    Else
      MkDir (strPath)
  End If
  strPath = strPath & "\"

  'fname = SelectFile(strPath, "*.txt", "filelist.txt", "Select the file names file")
  'fname = SelectFile(strPath, "*.txt", "key-ins.txt")
  'MsgBox strPath
End Sub

Sub testselectFolder()
  Dim T As String
  T = SelectDGNFolder
  T = T & "did it show up"
  MsgBox T
End Sub

Function SelectDGNFolder() As String
  'This function calls the SelectFile function listed below

  Dim fname As String
  Dim strPath As String
  Dim PPath As String
  Dim TitleInfo As String
  Dim AFolderName As String
  Dim BSlash As Integer

  TitleInfo = "To Select a Folder, Select a File inside of a Folder!"

  PPath = "P:\Active Projects\PGE\Substation\"

  strPath = PPath      'uses function in this module, string path could be hard coded

  fname = SelectFile(strPath, "*.dgn", "To select a folder, select a file inside a folder", TitleInfo)
  'MsgBox fname

  If fname = "" Then
    MsgBox "you did not select a file!"
    'clear the list box
    frmNextFile.ListBox1.Clear
    Exit Function
  End If

  BSlash = RightMostBackSlash(fname)

  AFolderName = FolderName(BSlash, fname)
  SelectDGNFolder = AFolderName

End Function
Function SelectFile(strStartingPath As String, strFilter As String, strSuggFName As String, TitleText As String) As String
  'this subroutine requires the declaration statement at the top of this module
  'This routine uses the function SelectDGNFolder listed above
  Dim strFName As String
  Dim lngfhandle As Long
  Dim lngrid As Long
  Dim retVal As Long
  Dim strPath As String
  strFName = Space(255)
  retVal = mdlDialog_fileOpen(filename:=strFName, rFileH:=lngfhandle, resourceId:=lngrid, _
                              suggestedFileName:=strSuggFName, filterString:=strFilter, defaultDirectory:=strStartingPath, _
                              titleString:=TitleText)

  Select Case retVal
     Case 0  'Open
       strFName = Left(strFName, InStr(1, strFName, Chr(0)) - 0.1)
       'MsgBox "File Selected:" & vbCr & strFName
     Case 1  'Cancel
       MsgBox "No File Selected."
       strFName = ""
     End Select
  SelectFile = strFName
End Function
Function RightMostBackSlash(strPath As String) As Integer
  'This function is used by the SelectDGNFolder listed above
  Dim count As Integer
  Dim LeftPart As String
  Dim RightPart As String
  count = 0

  'find the right most "\" backslash
  While LeftPart <> "\"
    count = count + 1
    RightPart = Right(strPath, count)
    'Debug.Print RightPart
    LeftPart = Left(RightPart, 1)
    'Debug.Print LeftPart
  Wend
  'MsgBox Str(count)
  RightMostBackSlash = count

End Function
Function FolderName(BackSlashPos As Integer, PathAndFileName As String) As String
  'I don't think this function will be used
  Dim Folderpath As String
  Dim filename As String
  Dim intBSlashLoc As Integer

  Trim (PathAndFileName)
  slen = Len(PathAndFileName)

  Folderpath = Left(PathAndFileName, slen - BackSlashPos + 1)
  'FileName = Right(PathAndFileName, intBSlashLoc - 1)
  'MsgBox "Folder Path:" & " " & folderPath & vbLf & "File Name:" & " " & FileName

  FolderName = Folderpath
End Function

Sub test_FileCreate()
   Call FileCreate("C:\Users\knowles_keith\Desktop\Microstation_test_Folder\")
End Sub

Function GetFileListPath() As String
   Dim DesktopPath As String
   Dim objFolders As Object

   Set objFolders = CreateObject("wScript.Shell").specialfolders
   DesktopPath = objFolders("desktop")
   GetFileListPath = DesktopPath & "\Filelists\filelist.txt"

End Function

Sub FileCreate(Folder As String)
  Dim textfile As String
  Dim Folderpath As String
  Dim DesktopPath As String
  Dim objFolders As Object

   Set objFolders = CreateObject("wScript.Shell").specialfolders
   DesktopPath = objFolders("desktop")

   textfile = GetFileListPath

   'folderPath = "C:\Users\knowles_keith\Desktop\Microstation_test_Folder\"
   'Folderpath = "C:\Users\knowles_keith\Desktop\Microstation_test_Folder\"

   Folderpath = Folder

   'folderPath = "P:\Active Projects\PGE\Substation\6446 BELL\2000 Substation\2300 Engineering\2310 Electrical\2311 Drawings\Indoor\"
   Call TextFileCreate(textfile, Folderpath)
```

```
End Sub

Sub TextFileCreate(textfile As String, Folderpath As String)
'*************************************************************
'Good - code is used in button, "Change Folder Path"

  Dim N As Integer
  Dim I As Integer
  Dim MyPath As String
  Dim MyName As String
  Dim FileNames() As String
  Dim counter As Integer

  MyPath = Folderpath

  'Display the names in the directory
    MyName = Dir(MyPath)    'Retrieve the first entry.
    I = 0
    Do While MyName <> ""    ' Start the loop.
        ' Ignore the current directory and the encompassing directory.
        If MyName <> "." And MyName <> ".." Then
          If Right(MyName, 4) = ".dgn" Then
            'Debug.Print MyName    ' Display entry only if it
             I = I + 1
             ReDim Preserve FileNames(I)
             FileNames(I) = MyName
             Debug.Print FileNames(I)
          End If
        End If    ' it represents a directory.
        MyName = Dir    ' Get next entry.
    Loop

    'create textfile or over-write the existing file
    'textfile

    Open textfile For Output As #1  'over-writes and/or creates new file
    Print #1, FileNames(1)
    Close #1

    If UBound(FileNames) <= 1 Then
        Exit Sub
    End If
    counter = 2
      Open textfile For Append As #1
    For counter = 2 To UBound(FileNames)
      Print #1, FileNames(counter)
    Next
    Close #1

 '*****************************************************************************
End Sub

Sub filepathsTxtFile()  'needs to take in the file path name, may need to
  Dim textfile As String
  Dim N As Integer
  Dim I As Integer
  Dim MyPath As String
  Dim MyName As String
  Dim FolderString() As String

  'This folder should be passed to the routine by arguments
  Folderpath = "C:\Users\knowles_keith\Desktop\Microstation_test_Folder\"

  'get file --- usually located on the desktop folder with the name filelist.txt
  'textfile = "c:\filelist.txt"

  MyPath = Folderpath

  'Display the names in the directory
    MyName = Dir(MyPath)    ' Retrieve the first entry
    Do While MyName <> ""    ' Start the loop
        ' Ignore the current directory and the encompassing directory
        If MyName <> "." And MyName <> ".." Then
          If Right(MyName, 4) = ".dgn" Then
            Debug.Print MyName    ' Display entry only if it
          End If
        End If    ' it represents a directory.
        MyName = Dir    ' Get next entry.
    Loop
End Sub

Sub test_SplitFolderFromFileName()

  Dim Folderpath As String
  Dim filename As String
  Dim strPath As String
  Dim intBSlashLoc As Integer

  strPath = "C:\Indoor\123.dgn"

  Trim (strPath)
  slen = Len(strPath)

  'Get the position of the right most back slash in file path
  intBSlashLoc = RightMostBackSlash(strPath)

  Folderpath = Left(strPath, slen - intBSlashLoc + 1)
  filename = Right(strPath, intBSlashLoc - 1)
  MsgBox "Folder Path:" & " " & Folderpath & vbLf & "File Name:" & " " & filename

End Sub

Function WritePathToFile(FullFolderPath As String) As Boolean
   Dim textfilepath As String
   Dim path As String
   Dim IsFolderThere As Boolean

   'get the desktop folder path
   path = DesktopPath
   path = path & "\Filelists"

   'check to see if folder on desktop
   If (Dir(path, vbDirectory) <> "") Then
        IsFolderThere = True
        MsgBox "folder there"
     Else
        IsFolderThere = False
        MsgBox "Folder not there"
        MkDir (path)
   End If

   path = path & "\path.txt"

   Open path For Output As #1
      Print #1, FullFolderPath
   Close #1

End Function

Function GetPathToFile() As String
   Dim textfilepath As String
   Dim path As String
   Dim IsFolderThere As Boolean

   'get the desktop folder path
   path = DesktopPath
   path = path & "\Filelists"

   'check to see if folder on desktop
   If (Dir(path, vbDirectory) <> "") Then
        IsFolderThere = True
        'MsgBox "folder there"
     Else
        IsFolderThere = False
        'MsgBox "Folder not there"
        MkDir (path)
   End If

   path = path & "\path.txt"

   Open path For Input As #1
      Line Input #1, FullFolderPath
   Close #1

   GetPathToFile = FullFolderPath
End Function
```

```
Sub Macro1()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

    Dim modalHandler As New Macro1ModalHandler5
    AddModalDialogEventsHandler modalHandler

'   The following statement opens modal dialog "Preferences [descartes]"

'   Start a command
    CadInputQueue.SendCommand "MDL SILENTLOAD USERPREF"

    CadInputQueue.SendCommand "MDL SILENTUNLOAD SPELLCHECK"

    RemoveModalDialogEventsHandler modalHandler
    CommandState.StartDefaultCommand
End Sub
Sub Macro2()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

    Dim modalHandler As New Macro2ModalHandler1
    AddModalDialogEventsHandler modalHandler

'   The following statement opens modal dialog "Preferences [descartes]"

'   Start a command
    CadInputQueue.SendCommand "MDL SILENTLOAD USERPREF"

    CadInputQueue.SendCommand "MDL SILENTUNLOAD SPELLCHECK"

    RemoveModalDialogEventsHandler modalHandler
    CommandState.StartDefaultCommand
End Sub
Sub Macro3()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "DIALOG PLOT"

    Dim modalHandler As New Macro3ModalHandler
    AddModalDialogEventsHandler modalHandler

'   The following statement opens modal dialog "Print - Raster Options"

    CadInputQueue.SendCommand "PRINT ROPTSDIALOG"

    RemoveModalDialogEventsHandler modalHandler
    CommandState.StartDefaultCommand
End Sub
Sub Macro4()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Send a keyin that can be a command string
    CadInputQueue.SendKeyin "level purge all"

    Dim modalHandler As New Macro4ModalHandler1
    AddModalDialogEventsHandler modalHandler

'   The following statement opens modal dialog "Design File Settings"

'   Start a command
    CadInputQueue.SendCommand "MDL SILENTLOAD DGNSET"

    CadInputQueue.SendCommand "FILEDESIGN"
    CadInputQueue.SendCommand "MDL SILENTUNLOAD DGNSET"

    RemoveModalDialogEventsHandler modalHandler
    CommandState.StartDefaultCommand
End Sub
Sub Macro5()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

    Dim modalHandler As New Macro5ModalHandler0
    AddModalDialogEventsHandler modalHandler

'   The following statement opens modal dialog "Color Table"

'   Start a command
    CadInputQueue.SendCommand "DIALOG COLOR"

'   Coordinates are in master units
    startPoint.x = 2.95957877203563
    startPoint.Y = 0.120543355820554
    startPoint.Z = 0.083333333333315

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 0.178317424247461
    point.Y = startPoint.Y + 0.414411330316334
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "DELETE ELEMENT"

    point.x = startPoint.x - 0.111822281528084
    point.Y = startPoint.Y - 5.83859518712345E-02
    point.Z = startPoint.Z - 2.3592E-16
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 0.241407279381207
    point.Y = startPoint.Y + 0.130217249126066
    point.Z = startPoint.Z - 2.3592E-16
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "DELETE ELEMENT"

    CadInputQueue.SendCommand "MDL SILENTUNLOAD VBAPM"

'   The following statement opens modal dialog "Color Table"

    CadInputQueue.SendCommand "DIALOG COLOR"

'   The following statement opens modal dialog "Design File Settings"

    CadInputQueue.SendCommand "MDL SILENTLOAD DGNSET"

    CadInputQueue.SendCommand "FILEDESIGN"

    CadInputQueue.SendCommand "MDL SILENTUNLOAD DGNSET"

'   The following statement opens modal dialog "Level/Filter Import"

'   The following statement opens modal dialog "Import Levels"

    CadInputQueue.SendCommand "LEVELMANAGER LIBRARY IMPORT"

    point.x = startPoint.x + 0.40227260475232
    point.Y = startPoint.Y - 0.312869042962989
    point.Z = startPoint.Z + 0#
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 0.713249564377427
    point.Y = startPoint.Y - 2.95800476295584E-02
    point.Z = startPoint.Z + 0#
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

'   Set a variable associated with a dialog box
    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    point.x = startPoint.x - 2.55371618863213
    point.Y = startPoint.Y + 1.78409905360514
    point.Z = startPoint.Z + 1.80411E-15
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 0.213079974691378
    point.Y = startPoint.Y - 0.414583282239851
    point.Z = startPoint.Z + 1.80411E-15
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 3.5085772346024
    point.Y = startPoint.Y + 2.19851038392148
    point.Z = startPoint.Z + 1.80411E-15
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 0.207327799715654
    point.Y = startPoint.Y - 0.35702615302925
    point.Z = startPoint.Z + 1.80411E-15
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "Change Attributes"

'   Send a keyin that can be a command string
    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES USEACTIVE ON"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES ENABLE LEVEL"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET LEVEL ""New or Revisions"""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES ENABLE COLOR"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET COLOR ""0"""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE LINESTYLE"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET LINESTYLE ""Continuous"""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE WEIGHT"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET WEIGHT 0"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE TRANSPARENCY"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET TRANSPARENCY 0"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE PRIORITY"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET PRIORITY 0"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE ELEMENTCLASS"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET ELEMENTCLASS PRIMARY"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE TEMPLATE"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET TEMPLATE """""
    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES MAKECOPY OFF"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES ENTIREELEMENT OFF"

    SetCExpressionValue "tcb->msToolSettings.general.useFence", 0, "CHANGEATTRIBS"

    CadInputQueue.SendCommand "LOCK FENCE INSIDE"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET LEVEL ""Vendor"""

    point.x = startPoint.x + 0.115293000104061
    point.Y = startPoint.Y + 0.926497828367175
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    point.x = startPoint.x - 3.20191440620909
    point.Y = startPoint.Y + 1.94525901539483
    point.Z = startPoint.Z + 2.34535E-15
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 0.631411213932183
    point.Y = startPoint.Y - 6.28739690214429E-03
    point.Z = startPoint.Z + 2.34535E-15
    CadInputQueue.SendDataPoint point, 1

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    CadInputQueue.SendCommand "NEWFILE U:\New folder\bellSWGRTITLEBLOCK.dgn"

    CadInputQueue.SendKeyin "task sendtaskchangedasync"

    CadInputQueue.SendKeyin "task sendtaskchangedasync ""\Drawing"""

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    point.x = startPoint.x - -39.6470393651277
    point.Y = startPoint.Y - 5.9953998274684
    point.Z = startPoint.Z + 2.45666666666633
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -17.8879261258157
    point.Y = startPoint.Y + 13.083591904566
    point.Z = startPoint.Z + 2.45666666666633
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "MDL LOAD CLIPBRD COPY"

    CadInputQueue.SendCommand "NEWFILE ""P:\Active Projects\PGE\Substation\6446 BELL\2000 Substation\2300 Engineering\2310 Electrical\2311 Drawings\Indoor\bell7313a0.dgn"",""~4683"""

    CadInputQueue.SendKeyin "task sendtaskchangedasync"

    CadInputQueue.SendKeyin "task sendtaskchangedasync ""\Drawing"""

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    CadInputQueue.SendCommand "MDL KEYIN CLIPBRD CLIPBOARD PASTE"

    point.x = startPoint.x - 28.631953394003
    point.Y = startPoint.Y + 16.2454221816907
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

'   Send a reset to the current command
    CadInputQueue.SendReset

    point.x = startPoint.x - -9.72067325664033
    point.Y = startPoint.Y - 10.5493999921612
    point.Z = startPoint.Z + 2.45666666666622
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "MDL SILENTLOAD USTNVBA MACROS"

    CadInputQueue.SendReset

    point.x = startPoint.x - 6.70546566878871
    point.Y = startPoint.Y + 56.2127948024225
    point.Z = startPoint.Z + 2.45666666666622
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 93.0284853312103
    point.Y = startPoint.Y - 26.450259538265
    point.Z = startPoint.Z + 2.45666666666622
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "SCALE ICON"

    CadInputQueue.SendCommand "ACTIVE XSCALE 0.3900"

    CadInputQueue.SendCommand "ACTIVE SCALE"

    point.x = startPoint.x - -15.9807978633999
    point.Y = startPoint.Y - 9.31802029667172
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "MOVE ICON"

'   Send a tentative point
    CadInputQueue.SendTentativePoint Point3dFromXYZ(43.3222605049795, 15.6762988369673, 2.53999999999963), 1

    point.x = startPoint.x - -40.2932509755801
    point.Y = startPoint.Y + 15.5668957102602
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(-10.8485751025206, 26.7805644149428, 0#), 1

    point.x = startPoint.x - 13.757003394003
    point.Y = startPoint.Y + 26.7454221816907
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    point.x = startPoint.x - 17.8581020925771
    point.Y = startPoint.Y + 6.35711818150786
    point.Z = startPoint.Z + 2.45666666666631
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 18.0026820661479
    point.Y = startPoint.Y + 6.73084617122634
    point.Z = startPoint.Z + 2.45666666666631
    CadInputQueue.SendDataPoint point, 1

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 2, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR MODE REMOVE"

    point.x = startPoint.x - 19.2263406966295
    point.Y = startPoint.Y + 5.90013022633797
    point.Z = startPoint.Z + 2.45666666666631
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 19.4851915607698
    point.Y = startPoint.Y + 7.26581970489595
    point.Z = startPoint.Z + 2.45666666666631
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "MOVE ICON"

    point.x = startPoint.x - 18.457003394003
    point.Y = startPoint.Y + 6.8954221816907
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 17.7338536777898
    point.Y = startPoint.Y + 7.65405225873153
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "MOVE ICON"

    point.x = startPoint.x - 17.4179055416471
    point.Y = startPoint.Y + 7.88136182113021
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 17.0881472557046
    point.Y = startPoint.Y + 6.44033105421862
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    point.x = startPoint.x - 18.0520137461763
    point.Y = startPoint.Y + 5.93398990685809
    point.Z = startPoint.Z + 2.45666666666631
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 22.0994999854615
    point.Y = startPoint.Y + 11.4532591098579
    point.Z = startPoint.Z + 2.45666666666631
    CadInputQueue.SendDataPoint point, 1

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 1, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR MODE ADD"

    point.x = startPoint.x - 17.9014096070401
    point.Y = startPoint.Y + 5.91515281060212
    point.Z = startPoint.Z + 2.45666666666631
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 22.1936275724216
    point.Y = startPoint.Y + 11.2837252435542
    point.Z = startPoint.Z + 2.45666666666631
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "MOVE ICON"

    point.x = startPoint.x - 21.8924192941492
    point.Y = startPoint.Y + 11.1330284735064
    point.Z = startPoint.Z + 2.45666666666631
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 21.9213352888634
    point.Y = startPoint.Y + 11.8660175630188
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    CadInputQueue.SendCommand "EDIT SINGLE DIALOG"

    point.x = startPoint.x - 12.979263944635
    point.Y = startPoint.Y + 8.66307073823081
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

'   Send a message string to an application
'   Content is defined by the application
    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine BELL SUBSTATION #3                                      "

    point.x = startPoint.x - 12.979263944635
    point.Y = startPoint.Y + 8.66307073823081
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - 12.8346839710643
    point.Y = startPoint.Y + 8.8197953790805
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine 15KV OUTDOOR SWITCHGEAR                                      "

    point.x = startPoint.x - 12.8105873088025
    point.Y = startPoint.Y + 8.8197953790805
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - 12.4491373748756
    point.Y = startPoint.Y + 8.85596260389197
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine UNIT 4 - FDR. BKR R304 CONTROL SCHEM.                                      "

    point.x = startPoint.x - 12.4491373748756
    point.Y = startPoint.Y + 8.85596260389197
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - 11.1479176127389
    point.Y = startPoint.Y + 8.94035279511872
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 12.7764830014951
    point.Y = startPoint.Y + 7.54338620193733
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine 3 "

    point.x = startPoint.x - 12.7764830014951
    point.Y = startPoint.Y + 7.54338620193733
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - 13.3702247596256
    point.Y = startPoint.Y + 7.41221973328774
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine BELL-7313"

    point.x = startPoint.x - 14.4343333651063
    point.Y = startPoint.Y + 7.12673977210923
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 5, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR ALL"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    CadInputQueue.SendKeyin "VBA RUN BUTTONS"

    point.x = startPoint.x - 57.5043080250155
    point.Y = startPoint.Y + 33.4859425904581
    point.Z = startPoint.Z + 2.45666666666631
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 8.85056034016884
    point.Y = startPoint.Y - -0.431345583671012
    point.Z = startPoint.Z + 2.45666666666631
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(-44.1353407261628, 5.16875384750476, 2.53999999999963), 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(-43.043773250429, 4.99629508051283, 2.53999999999963), 1

    point.x = startPoint.x - 45.757003394003
    point.Y = startPoint.Y + 5.2454221816907
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendAdjustedDataPoint point, 1

    point.x = startPoint.x - 48.0141342146059
    point.Y = startPoint.Y + 7.92252327488308
    point.Z = startPoint.Z + 2.45666666666631
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendKeyin Chr$(27)

    CadInputQueue.SendCommand "MOVE ICON"
    CadInputQueue.SendTentativePoint Point3dFromXYZ(-42.9458768831316, 4.95628464657069, 2.53999999999963), 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(-42.7105579157313, 5.42721205296999, 2.53999999999963), 1

    point.x = startPoint.x - 45.757003394003
    point.Y = startPoint.Y + 5.2454221816907
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendKeyin "xy=0,0"

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "VIEW ON 5"

    CadInputQueue.SendKeyin "dialog viewsettings popup"

    CadInputQueue.SendKeyin "MDL KEYIN BENTLEY.VIEWATTRIBUTESDIALOG,VAD VIEWATTRIBUTESDIALOG SETATTRIBUTE 0 DataFields False"

    point.x = startPoint.x - -31.2800461097629
    point.Y = startPoint.Y + 4.42010573726455
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -32.3080604127794
    point.Y = startPoint.Y + 6.25881152715817
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "PRINT EXECUTE"

    point.x = startPoint.x - -32.1410080885392
    point.Y = startPoint.Y + 6.49025701120072
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    point.x = startPoint.x - -31.7426525461203
    point.Y = startPoint.Y + 6.2973857744986
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

'   The following statement opens modal dialog "Open"

    CadInputQueue.SendCommand "DIALOG OPENFILE"

    point.x = startPoint.x - -28.1960032007134
    point.Y = startPoint.Y + 4.6386931388603
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    point.x = startPoint.x - -27.3221910431493
    point.Y = startPoint.Y - -2.54282570003051
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

'   The following statement opens modal dialog "Open"

    CadInputQueue.SendCommand "DIALOG OPENFILE"
    CadInputQueue.SendKeyin "task sendtaskchangedasync"

    CadInputQueue.SendKeyin "task sendtaskchangedasync ""\Drawing"""

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    point.x = startPoint.x + 0.193136870243662
    point.Y = startPoint.Y - 0.277476094811245
    point.Z = startPoint.Z + 0#
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 0.285178612605209
    point.Y = startPoint.Y + 0.491060597448877
    point.Z = startPoint.Z + 0#
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "DELETE ELEMENT"

'   The following statement opens modal dialog "Design File Settings"

    CadInputQueue.SendCommand "MDL SILENTLOAD DGNSET"

    CadInputQueue.SendCommand "MDL SILENTUNLOAD DGNSET"

'   The following statement opens modal dialog "Color Table"

    CadInputQueue.SendCommand "DIALOG COLOR"

    point.x = startPoint.x - 3.6622037756937
    point.Y = startPoint.Y + 2.70297650225928
    point.Z = startPoint.Z + 0#
    CadInputQueue.SendDataPoint point, 1

'   The following statement opens modal dialog "Design File Settings"

    CadInputQueue.SendCommand "MDL SILENTLOAD USTNVBA MACROS"

    CadInputQueue.SendCommand "FILEDESIGN"

    CadInputQueue.SendCommand "MDL SILENTUNLOAD DGNSET"

    CadInputQueue.SendKeyin "level purge all"

'   The following statement opens modal dialog "Design File Settings"

    CadInputQueue.SendCommand "MDL SILENTLOAD DGNSET"

    CadInputQueue.SendCommand "FILEDESIGN"

    CadInputQueue.SendCommand "MDL SILENTUNLOAD DGNSET"

'   The following statement opens modal dialog "Alert"

    CadInputQueue.SendCommand "UNDO ALL"

    point.x = startPoint.x + 1.02681143702998
    point.Y = startPoint.Y + 0.608699634550071
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 0.99088028597463
    point.Y = startPoint.Y + 0.698582761919565
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 1.04477701255766
    point.Y = startPoint.Y + 0.761500951078211
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    point.x = startPoint.x + 0.936983559391602
    point.Y = startPoint.Y + 0.700380424466956
    point.Z = startPoint.Z + 0#
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 0.784276167406347
    point.Y = startPoint.Y + 0.853181740995096
    point.Z = startPoint.Z + 0#
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 0.125320921559938
    point.Y = startPoint.Y - 0.256515349508679
    point.Z = startPoint.Z + 0#
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 0.280543494119067
    point.Y = startPoint.Y + 0.117398460348417
    point.Z = startPoint.Z + 0#
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "DELETE ELEMENT"

'   The following statement opens modal dialog "Color Table"

    CadInputQueue.SendCommand "DIALOG COLOR"

    CadInputQueue.SendKeyin "level purge all"

    point.x = startPoint.x - 2.81984285313975
    point.Y = startPoint.Y + 1.81724569437845
    point.Z = startPoint.Z - 1.388E-17
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 0.672215890052462
    point.Y = startPoint.Y - 1.3736053272386
    point.Z = startPoint.Z - 1.388E-17
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendKeyin Chr$(27)

    CadInputQueue.SendCommand "SCALE ICON"

    CadInputQueue.SendCommand "ACTIVE XSCALE 0.3900"

    CadInputQueue.SendCommand "ACTIVE SCALE"

    point.x = startPoint.x - 2.00016346968949
    point.Y = startPoint.Y - 0.800600390258071
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

'   The following statement opens modal dialog "Design File Settings"

    CadInputQueue.SendCommand "MDL SILENTLOAD DGNSET"

    CadInputQueue.SendCommand "FILEDESIGN"

    CadInputQueue.SendCommand "MDL SILENTUNLOAD DGNSET"
```

```
'   The following statement opens modal dialog "Color Table"

    CadInputQueue.SendCommand "DIALOG COLOR"

    point.x = startPoint.x - 0.385769666542224
    point.Y = startPoint.Y - 0.422631053694063
    point.Z = startPoint.Z + 0#
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 0.459356663903589
    point.Y = startPoint.Y - 0.290092989400102
    point.Z = startPoint.Z + 0#
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    point.x = startPoint.x - 2.44252624279237
    point.Y = startPoint.Y + 0.265830558055125
    point.Z = startPoint.Z + 0#
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 0.830971000578482
    point.Y = startPoint.Y - 0.919648794796418
    point.Z = startPoint.Z + 0#
    CadInputQueue.SendDataPoint point, 1

'   The following statement opens modal dialog "Level/Filter Import"

'   The following statement opens modal dialog "Import Levels"

    CadInputQueue.SendCommand "LEVELMANAGER LIBRARY IMPORT"

    point.x = startPoint.x - 0.301144619576654
    point.Y = startPoint.Y - 0.595666859855624
    point.Z = startPoint.Z + 0#
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "Change Attributes"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES USEACTIVE ON"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES ENABLE LEVEL"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET LEVEL ""Vendor"""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES ENABLE COLOR"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET COLOR ""0"""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE LINESTYLE"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET LINESTYLE ""Continuous"""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE WEIGHT"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET WEIGHT 0"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE TRANSPARENCY"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET TRANSPARENCY 0"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE PRIORITY"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET PRIORITY 0"
    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE ELEMENTCLASS"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET ELEMENTCLASS PRIMARY"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE TEMPLATE"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET TEMPLATE """""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES MAKECOPY OFF"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES ENTIREELEMENT OFF"

    SetCExpressionValue "tcb->msToolSettings.general.useFence", 0, "CHANGEATTRIBS"

    CadInputQueue.SendCommand "LOCK FENCE INSIDE"

    point.x = startPoint.x - 0.551340410605294
    point.Y = startPoint.Y - 0.367406860238246
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    point.x = startPoint.x - 2.21647618527293
    point.Y = startPoint.Y + 0.190587594471574
    point.Z = startPoint.Z + 7.0777E-16
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 1.18165903487874
    point.Y = startPoint.Y - 0.913896274644771
    point.Z = startPoint.Z + 7.0777E-16
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    CadInputQueue.SendCommand "NEWFILE U:\New folder\bellSWGRTITLEBLOCK.dgn"

    CadInputQueue.SendKeyin "task sendtaskchangedasync"

    CadInputQueue.SendKeyin "task sendtaskchangedasync ""\Drawing"""

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    point.x = startPoint.x - 6.22655504146303
    point.Y = startPoint.Y + 26.8878035695086
    point.Z = startPoint.Z + 2.45666666666633
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -37.5159921922054
    point.Y = startPoint.Y - 8.57667517944953
    point.Z = startPoint.Z + 2.45666666666633
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "MDL LOAD CLIPBRD COPY"

    point.x = startPoint.x - -33.2538978463608
    point.Y = startPoint.Y - 11.6068679839491
    point.Z = startPoint.Z + 2.45666666666633
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -32.6930959587497
    point.Y = startPoint.Y - 10.5968037157826
    point.Z = startPoint.Z + 2.45666666666633
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -25.5148317973271
    point.Y = startPoint.Y - 11.1579505314307
    point.Z = startPoint.Z + 2.45666666666633
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "NEWFILE P:\Active Projects\PGE\Substation\6446 BELL\2000 Substation\2300 Engineering\2310 Electrical\2311 Drawings\Indoor\bell7313b0.dgn"

    CadInputQueue.SendKeyin "task sendtaskchangedasync"

    CadInputQueue.SendKeyin "task sendtaskchangedasync ""\Drawing"""

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    CadInputQueue.SendCommand "MDL KEYIN CLIPBRD CLIPBOARD PASTE"

    CadInputQueue.SendCommand "ACTIVE ANGLE 0.0000°"

    CadInputQueue.SendCommand "ACTIVE ANGLE"

    CadInputQueue.SendCommand "ACTIVE XSCALE 1.0000"

    CadInputQueue.SendCommand "ACTIVE SCALE"

    point.x = startPoint.x - 4.70169394288028
    point.Y = startPoint.Y - 1.99909755251633
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

    point.x = startPoint.x - -15.792365326435
    point.Y = startPoint.Y + 10.7901239732489
    point.Z = startPoint.Z + 2.45666666666614
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -51.9735448359856
    point.Y = startPoint.Y - 19.4560095039993
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "MOVE ICON"

    CadInputQueue.SendTentativePoint Point3dFromXYZ(49.5509086694719, 8.81060678696583, 2.53999999999945), 1

    point.x = startPoint.x - -46.5778380614257
    point.Y = startPoint.Y + 8.53319221428959
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(13.1579450193187, 8.54097098153532, 0#), 1

    point.x = startPoint.x - -10.1732560571197
    point.Y = startPoint.Y + 8.50090244748367
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    CadInputQueue.SendKeyin "VBA RUN BUTTONS"

    CadInputQueue.SendCommand "VIEW ON 5"
    CadInputQueue.SendCommand "PLACE FENCE ICON"

    point.x = startPoint.x - 25.7595787720356
    point.Y = startPoint.Y + 12.5794566441794
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    point.x = startPoint.x - 29.0959011922633
    point.Y = startPoint.Y + 11.981172036036
    point.Z = startPoint.Z - -2.45666666666612
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -15.9025063967995
    point.Y = startPoint.Y - 22.2194920782503
    point.Z = startPoint.Z - -2.45666666666612
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "MOVE ICON"

    point.x = startPoint.x - 21.8267439428803
    point.Y = startPoint.Y - 12.9990975525163
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendKeyin "xy=0,0"

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    CadInputQueue.SendCommand "NEWFILE U:\New folder\bellSWGRTITLEBLOCK.dgn"

    CadInputQueue.SendKeyin "task sendtaskchangedasync"

    CadInputQueue.SendKeyin "task sendtaskchangedasync ""\Drawing"""

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

'   The following statement opens modal dialog "Compress Options"

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

'   The following statement opens modal dialog "Level/Filter Import"

'   The following statement opens modal dialog "Import Levels"

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    point.x = startPoint.x - -31.410299380563
    point.Y = startPoint.Y + 4.80849955322854
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -30.6764212172336
    point.Y = startPoint.Y + 5.59931598527109
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -27.1952042886195
    point.Y = startPoint.Y - -1.43811523571386
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -26.7059521797332
    point.Y = startPoint.Y - -2.66199780911304
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "Change Attributes"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES USEACTIVE ON"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES ENABLE LEVEL"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET LEVEL ""Vendor"""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES ENABLE COLOR"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET COLOR ""0"""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE LINESTYLE"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET LINESTYLE ""Continuous"""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE WEIGHT"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET WEIGHT 0"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE TRANSPARENCY"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET TRANSPARENCY 0"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE PRIORITY"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET PRIORITY 0"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE ELEMENTCLASS"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET ELEMENTCLASS PRIMARY"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE TEMPLATE"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET TEMPLATE """""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES MAKECOPY OFF"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES ENTIREELEMENT OFF"

    SetCExpressionValue "tcb->msToolSettings.general.useFence", 0, "CHANGEATTRIBS"

    CadInputQueue.SendCommand "LOCK FENCE INSIDE"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET LEVEL ""Border-titleblock"""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE COLOR"

    point.x = startPoint.x - -34.7268641571481
    point.Y = startPoint.Y + 8.65431517810213
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"
    CadInputQueue.SendKeyin "level purge all"

    point.x = startPoint.x - -37.7780284085457
    point.Y = startPoint.Y - -2.10891197072276
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -32.9542348920698
    point.Y = startPoint.Y - -3.44201723772157
    point.Z = startPoint.Z + 2.45666666666632
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

    point.x = startPoint.x - -31.9062083245021
    point.Y = startPoint.Y - 0.844605732886673
    point.Z = startPoint.Z + 2.45666666666631
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -27.8145977524912
    point.Y = startPoint.Y + 4.68605728214073
    point.Z = startPoint.Z + 2.45666666666631
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    CadInputQueue.SendCommand "NEWFILE P:\Active Projects\PGE\Substation\6446 BELL\2000 Substation\2300 Engineering\2310 Electrical\2311 Drawings\Indoor\bell7313b0.dgn"

    CadInputQueue.SendKeyin "task sendtaskchangedasync"

    CadInputQueue.SendKeyin "task sendtaskchangedasync ""\Drawing"""

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    CadInputQueue.SendCommand "EDIT SINGLE DIALOG"

    point.x = startPoint.x - -29.4404212279644
    point.Y = startPoint.Y - -1.27945664417945
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine BELL-7313"

    point.x = startPoint.x - -31.7404212279644
    point.Y = startPoint.Y - -0.779456644179446
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - -30.0404212279644
    point.Y = startPoint.Y - -1.77945664417945
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine 21"

    point.x = startPoint.x - -30.0404212279644
    point.Y = startPoint.Y - -1.77945664417945
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - -29.7404212279644
    point.Y = startPoint.Y - -3.07945664417945
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine BELL SUBSTATION #3                                      "

    point.x = startPoint.x - -29.5404212279644
    point.Y = startPoint.Y + 3.67945664417945
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - -30.0404212279644
    point.Y = startPoint.Y + 3.57945664417945
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine 15KV OUTDOOR SWITCHGEAR                                      "

    point.x = startPoint.x - -30.0404212279644
    point.Y = startPoint.Y + 3.57945664417945
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - -30.3404212279644
    point.Y = startPoint.Y - -3.27945664417945
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine UNIT 4 - FDR. BKR R304 CONTROL SCHEM.                                      "

    point.x = startPoint.x - -30.3404212279644
    point.Y = startPoint.Y - -3.27945664417945
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - -31.2404212279644
    point.Y = startPoint.Y - -2.97945664417945
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendKeyin "dialog viewsettings popup"

    CadInputQueue.SendKeyin "MDL KEYIN BENTLEY.VIEWATTRIBUTESDIALOG,VAD VIEWATTRIBUTESDIALOG SETATTRIBUTE 0 DataFields False"

    CadInputQueue.SendKeyin "VBA RUN BUTTONS"

    CadInputQueue.SendCommand "PRINT EXECUTE"

    point.x = startPoint.x - -31.5756002218801
    point.Y = startPoint.Y + 6.28452769205179
    point.Z = startPoint.Z + 2.45666666666623
    CadInputQueue.SendDataPoint point, 1

    SetCExpressionValue "msDialogState.gridInfo.roundoffUnit", (ActiveModelReference.UORsPerMasterUnit * 0.05), "MGDSHOOK"

    CadInputQueue.SendCommand "ACTIVE UNITROUND"

    point.x = startPoint.x - -31.2800461097629
    point.Y = startPoint.Y + 4.80584821066881
    point.Z = startPoint.Z + 2.45666666666623
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

    point.x = startPoint.x - -28.8031741484325
    point.Y = startPoint.Y - -2.92696091312891
    point.Z = startPoint.Z + 2.45666666666622
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -26.650079006342
    point.Y = startPoint.Y - -2.52282284325535
    point.Z = startPoint.Z + 2.45666666666623
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -27.4721049433915
    point.Y = startPoint.Y - -1.33987925814897
    point.Z = startPoint.Z + 2.45666666666622
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -27.439208485695
    point.Y = startPoint.Y - -1.66904616878726
    point.Z = startPoint.Z + 2.45666666666622
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -27.2683011078185
    point.Y = startPoint.Y - -1.92929375751067
    point.Z = startPoint.Z + 2.45666666666622
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -27.3094216799392
    point.Y = startPoint.Y - -2.45583223370748
    point.Z = startPoint.Z + 2.45666666666622
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendKeyin Chr$(27)

    CadInputQueue.SendCommand "MOVE ICON"

    point.x = startPoint.x - -27.1904212279644
    point.Y = startPoint.Y - -2.17945664417945
    point.Z = startPoint.Z + 2.45666666666622
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -27.8404212279644
    point.Y = startPoint.Y - -1.57945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    point.x = startPoint.x - -22.7476082103034
    point.Y = startPoint.Y - -0.687594762848514
    point.Z = startPoint.Z + 2.45666666666622
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -21.366213990625
    point.Y = startPoint.Y + 4.56109209994958
    point.Z = startPoint.Z + 2.45666666666622
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "MOVE ICON"

    point.x = startPoint.x - -21.4883140025202
    point.Y = startPoint.Y + 4.29160664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -21.4404212279644
    point.Y = startPoint.Y + 5.52945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    CadInputQueue.SendCommand "PRINT EXECUTE"

    point.x = startPoint.x - -31.3956977188522
    point.Y = startPoint.Y + 6.1559468675837
    point.Z = startPoint.Z + 2.45666666666623
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -31.3956977188522
    point.Y = startPoint.Y + 6.16880495003051
    point.Z = startPoint.Z + 2.45666666666623
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -31.3956977188522
    point.Y = startPoint.Y + 6.18166303247732
    point.Z = startPoint.Z + 2.45666666666623
    CadInputQueue.SendDataPoint point, 1

'   The following statement opens modal dialog "Open"

    CadInputQueue.SendCommand "DIALOG OPENFILE"

    CadInputQueue.SendKeyin "task sendtaskchangedasync"

    CadInputQueue.SendKeyin "task sendtaskchangedasync ""\Drawing"""

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    point.x = startPoint.x - 7.39583659943892E-02
    point.Y = startPoint.Y - 8.62494185738905E-02
    point.Z = startPoint.Z - 6.10623E-15
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 0.363088088016569
    point.Y = startPoint.Y + 0.207879928787902
    point.Z = startPoint.Z - 6.10623E-15
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    point.x = startPoint.x - 7.87771946947586E-02
    point.Y = startPoint.Y - 0.115180174052099
    point.Z = startPoint.Z - 6.10623E-15
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 0.184791426102891
    point.Y = startPoint.Y + 0.101800492034469
    point.Z = startPoint.Z - 6.10623E-15
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "DELETE ELEMENT"

    point.x = startPoint.x - 0.160697282601043
    point.Y = startPoint.Y - 5.24968705159797E-02
    point.Z = startPoint.Z - 6.10623E-15
    CadInputQueue.SendDataPoint point, 1
    point.x = startPoint.x - 0.247436199207697
    point.Y = startPoint.Y + 9.21569068750659E-02
    point.Z = startPoint.Z - 6.10623E-15
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "DELETE ELEMENT"

    CadInputQueue.SendCommand "MDL SILENTLOAD USTNVBA MACROS"

'   The following statement opens modal dialog "Color Table"

    CadInputQueue.SendCommand "DIALOG COLOR"

    CadInputQueue.SendKeyin "level purge all"

    CadInputQueue.SendReset

'   The following statement opens modal dialog "Color Table"

    CadInputQueue.SendCommand "DIALOG COLOR"

    CadInputQueue.SendKeyin "level purge all"

    point.x = startPoint.x - 3.87185420612206
    point.Y = startPoint.Y - 0.87389677051904
    point.Z = startPoint.Z - 2.388367E-14
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 3.93067779865587
    point.Y = startPoint.Y - 0.638457679713302
    point.Z = startPoint.Z - 2.388367E-14
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

'   The following statement opens modal dialog "Level/Filter Import"

'   The following statement opens modal dialog "Import Levels"

    CadInputQueue.SendCommand "LEVELMANAGER LIBRARY IMPORT"

    point.x = startPoint.x - 3.61450098878665
    point.Y = startPoint.Y + 2.44358716836557
    point.Z = startPoint.Z - 3.001766E-14
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 3.22374164326869
    point.Y = startPoint.Y - 1.80535267351924
    point.Z = startPoint.Z - 3.001766E-14
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendKeyin Chr$(27)

    CadInputQueue.SendCommand "Change Attributes"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES USEACTIVE ON"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES ENABLE LEVEL"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET LEVEL ""Vendor"""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE COLOR"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET COLOR ""0"""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE LINESTYLE"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET LINESTYLE ""Continuous"""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE WEIGHT"
    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET WEIGHT 0"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE TRANSPARENCY"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET TRANSPARENCY 0"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE PRIORITY"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET PRIORITY 0"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE ELEMENTCLASS"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET ELEMENTCLASS PRIMARY"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES DISABLE TEMPLATE"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES SET TEMPLATE """""

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES MAKECOPY OFF"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES ENTIREELEMENT OFF"

    SetCExpressionValue "tcb->msToolSettings.general.useFence", 0, "CHANGEATTRIBS"

    CadInputQueue.SendCommand "LOCK FENCE INSIDE"

    CadInputQueue.SendKeyin "CHANGE ATTRIBUTES ENABLE COLOR"

    point.x = startPoint.x + 6.19735445764391E-02
    point.Y = startPoint.Y + 1.48711586196725
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

'   The following statement opens modal dialog "Design File Settings"

    CadInputQueue.SendCommand "MDL SILENTLOAD DGNSET"

    CadInputQueue.SendCommand "FILEDESIGN"

    CadInputQueue.SendCommand "MDL SILENTUNLOAD DGNSET"

    point.x = startPoint.x + 0.135503035243701
    point.Y = startPoint.Y - 0.77530665124414
    point.Z = startPoint.Z - 2.445266E-14
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 3.52258912545257
    point.Y = startPoint.Y + 3.51042054857907
    point.Z = startPoint.Z - 2.445266E-14
    CadInputQueue.SendDataPoint point, 1

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

'   The following statement opens modal dialog "Design File Settings"

    CadInputQueue.SendCommand "MDL SILENTLOAD DGNSET"

    CadInputQueue.SendCommand "FILEDESIGN"

    CadInputQueue.SendCommand "MDL SILENTUNLOAD DGNSET"

    CadInputQueue.SendCommand "NEWFILE U:\New folder\bellSWGRTITLEBLOCK.dgn"
    CadInputQueue.SendKeyin "task sendtaskchangedasync"

    CadInputQueue.SendKeyin "task sendtaskchangedasync ""\Drawing"""

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    point.x = startPoint.x - -33.6289917232435
    point.Y = startPoint.Y - 4.6783607773943
    point.Z = startPoint.Z + 2.4566666666663
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -21.0670294407541
    point.Y = startPoint.Y + 12.8294198708256
    point.Z = startPoint.Z + 2.4566666666663
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "MDL LOAD CLIPBRD COPY"

    CadInputQueue.SendCommand "NEWFILE ""P:\Active Projects\PGE\Substation\6446 BELL\2000 Substation\2300 Engineering\2310 Electrical\2311 Drawings\Indoor\bell7313c0.dgn"",""~9308"""

    CadInputQueue.SendKeyin "task sendtaskchangedasync"

    CadInputQueue.SendKeyin "task sendtaskchangedasync ""\Drawing"""

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    CadInputQueue.SendCommand "COMPONENTVIEW COMPONENTSETOVERRIDE SUSPEND"

    CadInputQueue.SendCommand "MDL KEYIN CLIPBRD CLIPBOARD PASTE"

    point.x = startPoint.x - 37.4892706296046
    point.Y = startPoint.Y + 30.9404508829546
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

    point.x = startPoint.x - 5.32835376563225
    point.Y = startPoint.Y + 69.9609485976437
    point.Z = startPoint.Z - -2.45666666666545
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 114.798833963212
    point.Y = startPoint.Y - 44.4094757385139
    point.Z = startPoint.Z - -2.45666666666545
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "SCALE ICON"

    CadInputQueue.SendCommand "ACTIVE XSCALE 0.3900"

    CadInputQueue.SendCommand "ACTIVE SCALE"

    point.x = startPoint.x + 105.834118461059
    point.Y = startPoint.Y + 28.249382075045
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "MOVE ICON"

    CadInputQueue.SendTentativePoint Point3dFromXYZ(98.2705269009586, 38.3111785306598, 2.5399999999987), 1

    point.x = startPoint.x + 95.1037765401523
    point.Y = startPoint.Y + 38.4830111570074
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendAdjustedDataPoint point, 1
    CadInputQueue.SendTentativePoint Point3dFromXYZ(-19.8272901125179, 41.5777451127749, 0#), 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(-19.7878628767806, 41.5777451127749, 0#), 1

    CadInputQueue.SendReset

    CadInputQueue.SendTentativePoint Point3dFromXYZ(-19.6892947874374, 41.5382936268357, 2.5399999999987), 1

    point.x = startPoint.x - 21.1149206296046
    point.Y = startPoint.Y + 41.4404508829546
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "MOVE ICON"

    CadInputQueue.SendTentativePoint Point3dFromXYZ(98.0953691155843, 38.4380474080459, 2.53999999999871), 1

    point.x = startPoint.x + 95.1037765401523
    point.Y = startPoint.Y + 38.4830111570074
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendCommand "LOCK ASSOCIATION OFF"

    CadInputQueue.SendCommand "LOCK UNIT ON"

    CadInputQueue.SendTentativePoint Point3dFromXYZ(-19.7282985463777, 41.402149975998, 0#), 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(-19.7282985463777, 41.402149975998, 0#), 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(-19.7282985463777, 41.402149975998, 0#), 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(-19.7899036022172, 41.5254358695581, 0#), 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(-19.7899036022172, 41.5254358695581, 0#), 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(-19.7899036022172, 41.5254358695581, 0#), 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(-19.7899036022172, 41.5254358695581, 0#), 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(-19.6520087280777, 41.5632255108319, 0#), 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(-19.6520087280777, 41.5632255108319, 0#), 1

    point.x = startPoint.x - 22.6143206296046
    point.Y = startPoint.Y + 41.4404508829546
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    point.x = startPoint.x - 66.0404762882776
    point.Y = startPoint.Y + 46.3408407936845
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 9.49248525517938
    point.Y = startPoint.Y + 6.9533530554312
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "MOVE ICON"
    point.x = startPoint.x - 54.6143206296046
    point.Y = startPoint.Y + 19.9404508829546
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendKeyin "xy=0,0"

    CadInputQueue.SendReset

    CadInputQueue.SendCommand "CHOOSE ELEMENT"

    SetCExpressionValue "powerSelectInfo.prefs.currMode", 4, "PSELECT"

    CadInputQueue.SendCommand "POWERSELECTOR DESELECT"

    CadInputQueue.SendKeyin "dialog viewsettings popup"

    CadInputQueue.SendKeyin "MDL KEYIN BENTLEY.VIEWATTRIBUTESDIALOG,VAD VIEWATTRIBUTESDIALOG SETATTRIBUTE 0 DataFields False"

    CadInputQueue.SendKeyin "dialog viewsettings popup"

    CadInputQueue.SendKeyin "MDL KEYIN BENTLEY.VIEWATTRIBUTESDIALOG,VAD VIEWATTRIBUTESDIALOG SETATTRIBUTE 0 DataFields True"

    CadInputQueue.SendCommand "EDIT SINGLE DIALOG"

    point.x = startPoint.x - -29.4404212279644
    point.Y = startPoint.Y - -2.07945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine BELL-XXXX"

    point.x = startPoint.x - -29.4404212279644
    point.Y = startPoint.Y - -2.07945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine BELL-7314"

    point.x = startPoint.x - -30.9404212279644
    point.Y = startPoint.Y - -1.57945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - -30.0404212279644
    point.Y = startPoint.Y - -2.27945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - -30.0404212279644
    point.Y = startPoint.Y - -2.17945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine 2 "

    point.x = startPoint.x - -30.0404212279644
    point.Y = startPoint.Y - -2.17945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - -29.9404212279644
    point.Y = startPoint.Y - -2.47945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1
    point.x = startPoint.x - -29.6404212279644
    point.Y = startPoint.Y - -3.27945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine BELL SUBSTATION #3                                      "

    point.x = startPoint.x - -29.6404212279644
    point.Y = startPoint.Y - -3.27945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - -30.0404212279644
    point.Y = startPoint.Y - -3.27945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine 15KV OUTDOOR SWITCHGEAR                                      "

    point.x = startPoint.x - -30.0404212279644
    point.Y = startPoint.Y - -3.27945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - -30.4404212279644
    point.Y = startPoint.Y - -3.17945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine UNIT 5 - MAIN BKR R302 CONTROL SCHEM.                                      "

    point.x = startPoint.x - -30.4404212279644
    point.Y = startPoint.Y - -3.17945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - -32.1404212279644
    point.Y = startPoint.Y - -2.37945664417945
    point.Z = startPoint.Z - 0.083333333333315
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendKeyin "dialog viewsettings popup"

    CadInputQueue.SendKeyin "MDL KEYIN BENTLEY.VIEWATTRIBUTESDIALOG,VAD VIEWATTRIBUTESDIALOG SETATTRIBUTE 0 DataFields False"

'   The following statement opens modal dialog "Design File Settings"

    CadInputQueue.SendCommand "MDL SILENTLOAD DGNSET"

    CadInputQueue.SendCommand "FILEDESIGN"

    CadInputQueue.SendCommand "MDL SILENTUNLOAD DGNSET"

'   The following statement opens modal dialog "Color Table"

    CadInputQueue.SendCommand "DIALOG COLOR"

    CadInputQueue.SendCommand "EXIT"

    CadInputQueue.SendCommand "PRINT EXIT PLOTDLG"

    RemoveModalDialogEventsHandler modalHandler
    CommandState.StartDefaultCommand
End Sub
Sub Macro6()
```

```
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

    Dim modalHandler As New Macro6ModalHandler
    AddModalDialogEventsHandler modalHandler

'   The following statement opens modal dialog "Print Attributes"

'   Start a command
    CadInputQueue.SendCommand "PRINT ATTRIBDIALOG"

    RemoveModalDialogEventsHandler modalHandler
    CommandState.StartDefaultCommand
End Sub
Sub Macro7()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

    CommandState.StartDefaultCommand
End Sub
Sub Macro8()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

    Dim modalHandler As New Macro8ModalHandler
    AddModalDialogEventsHandler modalHandler

'   The following statement opens modal dialog "Print Attributes"

'   Start a command
    CadInputQueue.SendCommand "PRINT ATTRIBDIALOG"

    RemoveModalDialogEventsHandler modalHandler
    CommandState.StartDefaultCommand
End Sub
Sub Macro9()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "PLACE FENCE ICON"

'   Send a tentative point
'   Coordinates are in master units
    CadInputQueue.SendTentativePoint Point3dFromXYZ(0.098639241090924, 14.4992497162524, 1.4111111111138), 1

'   Coordinates are in master units
    startPoint.x = 0#
    startPoint.Y = 14.6666666666666
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(22.6630469909838, -4.14684863753751E-03, 1.41111111111402), 1

    point.x = startPoint.x + 22.6666666666667
    point.Y = startPoint.Y - 14.6666666666666
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    point.x = startPoint.x + 23.1448617967697
    point.Y = startPoint.Y - 6.5622857142857
    point.Z = startPoint.Z + 1.4111111111123
    CadInputQueue.SendDataPoint point, 5

    CadInputQueue.SendCommand "FIT VIEW EXTENDED 5"

    CadInputQueue.SendCommand "WINDOW AREA EXTENDED 1"

    point.x = startPoint.x + 21.7245776434224
    point.Y = startPoint.Y - 8.34410349586891
    point.Z = startPoint.Z + 1.41111111111421
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 21.6370483654411
    point.Y = startPoint.Y - 14.7128887948723
    point.Z = startPoint.Z + 1.41111111111426
    CadInputQueue.SendDataPoint point, 1

    CommandState.StartDefaultCommand
End Sub

Sub Macro10()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "PLACE FENCE ICON"

'   Send a tentative point
'   Coordinates are in master units
    CadInputQueue.SendTentativePoint Point3dFromXYZ(0.365598749417673, 43.5815800805484, 4.23333333333436), 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(0.102272749327261, 44.0333269768638, 4.23333333333436), 1

'   Coordinates are in master units
    startPoint.x = 0#
    startPoint.Y = 43.9999999999999
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(67.9812786286272, 3.29940720802703E-02, 4.23333333333444), 1

    point.x = startPoint.x + 68#
    point.Y = startPoint.Y - 43.9999999999999
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    point.x = startPoint.x + 8.83333333333333
    point.Y = startPoint.Y + 56.1666666666667
    point.Z = startPoint.Z + 4.23333333333333
    CadInputQueue.SendDataPoint point, 5

    CadInputQueue.SendCommand "FIT VIEW EXTENDED 5"

    CadInputQueue.SendCommand "WINDOW AREA EXTENDED 1"

    point.x = startPoint.x + 65.6749694500775
    point.Y = startPoint.Y - 24.7501961741185
    point.Z = startPoint.Z + 4.23333333333444
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 65.7208898602495
    point.Y = startPoint.Y - 44.117079717327
    point.Z = startPoint.Z + 4.23333333333444
    CadInputQueue.SendDataPoint point, 1

    CommandState.StartDefaultCommand
```

```
End Sub
Sub Macro11()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Coordinates are in master units
    startPoint.x = -2.38719521710856
    startPoint.Y = 5.40902255639098
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 5

'   Send a keyin that can be a command string
    CadInputQueue.SendKeyin "dialog viewsettings popup"

    CadInputQueue.SendKeyin "MDL KEYIN BENTLEY.VIEWATTRIBUTESDIALOG,VAD VIEWATTRIBUTESDIALOG SETATTRIBUTE 4 DataFields False"

    CommandState.StartDefaultCommand
End Sub
Sub Macro12()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "PLACE FENCE ICON"

'   Send a tentative point
'   Coordinates are in master units
    CadInputQueue.SendTentativePoint Point3dFromXYZ(0.493826490298015, 30.0246028833885, 0#), 1

'   Coordinates are in master units
    startPoint.x = 0#
    startPoint.Y = 30#
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(41.97131768619, 1.17293206833481E-02, 0#), 1

    point.x = startPoint.x + 42.0000000000001
    point.Y = startPoint.Y - 30#
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    point.x = startPoint.x + 1.79290742839691
    point.Y = startPoint.Y + 3.46071529917275
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 5

    CadInputQueue.SendCommand "PRINT MAXIMIZE"

    CadInputQueue.SendCommand "WINDOW AREA EXTENDED 1"

    point.x = startPoint.x + 40.9646372074583
    point.Y = startPoint.Y - 30.3439460111004
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 40.801902248909
    point.Y = startPoint.Y - 20.3922465283965
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CommandState.StartDefaultCommand
End Sub
Sub Macro13()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "PLACE FENCE ICON"

'   Send a tentative point
'   Coordinates are in master units
    CadInputQueue.SendTentativePoint Point3dFromXYZ(-7.68330441137402, 29.5718401832303, 0#), 1

'   Coordinates are in master units
    startPoint.x = -8.00000000000005
    startPoint.Y = 30#
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    CadInputQueue.SendTentativePoint Point3dFromXYZ(33.9605377145485, 2.06433146363044E-02, 0#), 1

    point.x = startPoint.x + 42#
    point.Y = startPoint.Y - 30#
    point.Z = startPoint.Z
    CadInputQueue.SendAdjustedDataPoint point, 1

    point.x = startPoint.x - 4.59999999999996
    point.Y = startPoint.Y - 14.05
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 5

    CadInputQueue.SendCommand "FIT VIEW EXTENDED 5"

    CadInputQueue.SendCommand "WINDOW AREA EXTENDED 1"

    point.x = startPoint.x + 40.7100968867561
    point.Y = startPoint.Y - 20.425106943707
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 40.9609209462061
    point.Y = startPoint.Y - 30.0283192740974
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CommandState.StartDefaultCommand
End Sub
Sub Macro14()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Coordinates are in master units
    startPoint.x = -6.56545142075177
    startPoint.Y = 54.3998934213841
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1
    point.x = startPoint.x - 5.95906729282748
    point.Y = startPoint.Y + 4.23707626929689
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

'   Start a command
    CadInputQueue.SendCommand "PLACE FENCE ICON"

    point.x = startPoint.x - 0.439089168945145
    point.Y = startPoint.Y + 8.69041997317247
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 6.20997538936757
    point.Y = startPoint.Y + 13.4610539948993
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendCommand "PLACE FENCE ICON"

    point.x = startPoint.x + 0.354799730263835
    point.Y = startPoint.Y - 3.12043012749257
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 5.91790268323876
    point.Y = startPoint.Y + 4.72600740824241
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CommandState.StartDefaultCommand
End Sub
Sub Macro15()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Coordinates are in master units
    startPoint.x = -7.0696445597244
    startPoint.Y = 57.3839304543931
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 6.42324727142667
    point.Y = startPoint.Y + 6.62867043018892
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

'   Start a command
    CadInputQueue.SendCommand "ORDER ELEMENT FRONT"

    CommandState.StartDefaultCommand
End Sub
Sub Macro16()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Coordinates are in master units
    startPoint.x = 33.1527004748686
    startPoint.Y = 0.754570801973472
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

'   Send a message string to an application
'   Content is defined by the application
    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 16"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 3"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 16"

    point.x = startPoint.x + 2.02202552933635
    point.Y = startPoint.Y + 0.511180800000016
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 0.474140430326564
    point.Y = startPoint.Y + 11.9115087470084
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 27"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 3"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 27"

    point.x = startPoint.x + 0.491130115385928
    point.Y = startPoint.Y + 11.7356298362372
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 0.531604437698213
    point.Y = startPoint.Y + 12.735398608026
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 83"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 3"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 83"

    point.x = startPoint.x + 1.09579141498126
    point.Y = startPoint.Y + 13.787661622426
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 0.486300933930082
    point.Y = startPoint.Y + 18.8328585963654
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 3"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    point.x = startPoint.x + 0.966672293459965
    point.Y = startPoint.Y + 19.4093287712902
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 0.483975647140291
    point.Y = startPoint.Y + 19.2116423764102
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"
    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 3"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    point.x = startPoint.x - 0.465645794156544
    point.Y = startPoint.Y + 20.0158674528391
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 3"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    point.x = startPoint.x - 0.427927708092867
    point.Y = startPoint.Y + 19.6215636852625
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 3"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    point.x = startPoint.x - 0.435867267566309
    point.Y = startPoint.Y + 20.4314206033249
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 setColor 3"

    CadInputQueue.SendMessageToApplication "WORDPROC", "1 selection 13 18"

    point.x = startPoint.x - 0.274746316731445
    point.Y = startPoint.Y + 20.4222726665818
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CommandState.StartDefaultCommand
End Sub
Sub Macro17()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Coordinates are in master units
    startPoint.x = 32.5440839946826
    startPoint.Y = 13.6218545103832
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

'   Send a message string to an application
'   Content is defined by the application
    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine REVISED PRIOR TO CONSTRUCTION, AWO 1000001215.                                                                    "

    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 0.857899547846586
    point.Y = startPoint.Y - 0.754601892977787
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1, 2

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - 8.19649249535317E-02
    point.Y = startPoint.Y + 6.9992059638519
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 9.84945181524211E-02
    point.Y = startPoint.Y + 5.95595517648597
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine PES   "

    point.x = startPoint.x + 9.84945181524211E-02
    point.Y = startPoint.Y + 5.95595517648597
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x + 8.10086674956736E-02
    point.Y = startPoint.Y + 6.33216249704301
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine DDB   "

    point.x = startPoint.x + 8.10086674956736E-02
    point.Y = startPoint.Y + 6.33216249704301
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x + 5.91513541747446E-02
    point.Y = startPoint.Y + 6.75648935860153
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine RCL   "

    point.x = startPoint.x + 5.91513541747446E-02
    point.Y = startPoint.Y + 6.74774035114672
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x + 7.22657421673034E-02
    point.Y = startPoint.Y + 7.1501946940682
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine DDB   "

    point.x = startPoint.x + 7.22657421673034E-02
    point.Y = startPoint.Y + 7.1501946940682
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x + 8.53801301598622E-02
    point.Y = startPoint.Y + 7.52202751089783
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1
    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine REJ   "

    point.x = startPoint.x + 8.53801301598622E-02
    point.Y = startPoint.Y + 7.52202751089783
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x - 0.771426552020685
    point.Y = startPoint.Y + 7.2114377462519
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CommandState.StartDefaultCommand
End Sub
```

```
Sub Buttons()
  frmButtons.show vbModeless
End Sub


Sub D_BORDER()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "DIALOG CELLMAINTENANCE"

    Dim modalHandler As New Macro1ModalHandler3
    AddModalDialogEventsHandler modalHandler

'   The following statement opens modal dialog "Attach Cell Library"

    CadInputQueue.SendCommand "ATTACH LIBRARY"

'   Set a variable associated with a dialog box
    SetCExpressionValue "tcb->activeCell", "BDR-D10", ""

'   Send a keyin that can be a command string
    CadInputQueue.SendKeyin "inputmanager currenttask"

    CadInputQueue.SendCommand "INPUTMANAGER MENU -609 2"

    CadInputQueue.SendCommand "DMSG ACTIVATETOOLBYPATH \Drawing\Cells\Place Active Cell"

    CadInputQueue.SendCommand "PLACE CELL ICON"

    CadInputQueue.SendKeyin "xy=0,0"

'   Send a reset to the current command
    CadInputQueue.SendReset

    RemoveModalDialogEventsHandler modalHandler
    CommandState.StartDefaultCommand
End Sub
Sub CRTS()
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Start a command
    CadInputQueue.SendCommand "INPUTMANAGER MENU -609,7"

    CadInputQueue.SendCommand "DMSG ACTIVATETOOLBYPATH \Drawing\Text\Edit Text"

    CadInputQueue.SendCommand "EDIT TEXT"

'   Coordinates are in master units
    startPoint.x = 32.143094
    startPoint.Y = 4.92251
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

'   Send a message string to an application
'   Content is defined by the application
    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine CURTIS SUBSTATION"

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "NextLine "

    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine CURTIS SUBSTATION"

    point.x = startPoint.x + 1.191537
    point.Y = startPoint.Y - 2.839316
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine D-<<     >>"

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "NextLine "

'   Send a keyin that can be a command string
    CadInputQueue.SendKeyin Chr$(27)

    CadInputQueue.SendCommand "INPUTMANAGER MENU -609,7"

    CadInputQueue.SendCommand "DMSG ACTIVATETOOLBYPATH \Drawing\Text\Fill In Single Enter-Data Field"

    CadInputQueue.SendCommand "EDIT SINGLE DIALOG"

    point.x = startPoint.x + 1.173366
    point.Y = startPoint.Y - 2.799316
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine CON     "

    point.x = startPoint.x + 1.173366
    point.Y = startPoint.Y - 2.799316
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    point.x = startPoint.x + 0.144069
    point.Y = startPoint.Y + 10.062997
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine AWO 0000026594.                                                    "



    point.x = startPoint.x - 0.242137
    point.Y = startPoint.Y + 16.100209
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 0.268499
    point.Y = startPoint.Y - 3.622898
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine CRTS-         "

    point.x = startPoint.x + 0.268499
    point.Y = startPoint.Y - 3.622898
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    CadInputQueue.SendMessageToApplication "TEXTEDIT", "FirstLine "

    CommandState.StartDefaultCommand
End Sub



Sub Macro1()
```

```
    Dim startPoint As Point3d
    Dim point As Point3d, point2 As Point3d
    Dim lngTemp As Long

'   Set a variable associated with a dialog box
    SetCExpressionValue "plotUI.uiPlotArea", 2, "PLOTDLG"

'   Coordinates are in master units
    startPoint.x = 35.175694
    startPoint.Y = 4.473955
    startPoint.Z = 0#

'   Send a data point to the current command
    point.x = startPoint.x
    point.Y = startPoint.Y
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 0.463557
    point.Y = startPoint.Y - 0.42816
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

'   Start a command
    CadInputQueue.SendCommand "MDL SILENTLOAD USTNVBA IDE"

    SetCExpressionValue "plotUI.uiPlotArea", 3, "PLOTDLG"

'   Send a keyin that can be a command string
    CadInputQueue.SendKeyin "VBA RUN BUTTONS"

    point.x = startPoint.x + 2.448534
    point.Y = startPoint.Y + 1.35584
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 0.237722
    point.Y = startPoint.Y + 1.28448
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x + 2.115724
    point.Y = startPoint.Y + 2.652213
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1

    point.x = startPoint.x - 0.33281
    point.Y = startPoint.Y + 1.71264
    point.Z = startPoint.Z
    CadInputQueue.SendDataPoint point, 1
End Sub
```

```
Sub TestColorTable()
  Dim message As String
  message = ColorTable
  MsgBox message

End Sub

Function ColorTable() As String

    Dim ct As ColorTable
    Dim ArrayRGBLongs() As Long
    Dim color As Long
    Dim r As Byte, g As Byte, b As Byte
    Dim message As String

    message = ""

    'Get a copy of the colortable that is currently attached to the active design file
    Set ct = Application.ActiveDesignFile.ExtractColorTable

    Dim cIndex As Long

    'Get an array of all of the RGB color values
    ArrayRGBLongs = ct.GetColors

    '35 is the highest color number assigned to a level
    For cIndex = 0 To 35

      ExtractRGB ArrayRGBLongs(cIndex), r, g, b

      message = CheckColor(cIndex, r, g, b)

      If message <> "" Then
        ColorTable = message & "One or More Colors Off!"
        'get out of for loop early, Fix manually, by attaching the new Color Table
        Exit Function
      End If
    Next
    ColorTable = message
    If message <> "" Then
      MsgBox message
    End If

End Function

Public Sub ExtractRGB(ByVal longColor As Long, intRed As Byte, intGreen As Byte, intBlue As Byte)
    Dim lngColor As Long

    lngColor = longColor
    intRed = lngColor Mod &H100
    lngColor = lngColor \ &H100
    intGreen = lngColor Mod &H100
    lngColor = lngColor \ &H100
    intBlue = lngColor Mod &H100
End Sub

Function CheckColor(chkIndex As Long, r As Byte, g As Byte, b As Byte) As String
 Dim message As String
 Dim CheckIndex As Integer

 CheckIndex = CInt(chkIndex)

  message = ""

      Select Case CheckIndex

' 0 is:  (r = 255, g = 255, b = 255)
        Case 0
            If r <> 255 Then
               message = "Red for Color 0 should be 255 not:" & r & vbCr
            End If
            If g <> 255 Then
               message = message & "Green for color 0 not 255:" & g & vbCr
            End If
            If b <> 255 Then
               message = message & "Blue for color 0 should be 255 not: " & b & vbCr
            End If
               CheckColor = message
            Exit Function

' 1 is:  (r = 0, g = 0, b = 255)
        Case 1
                If r <> 0 Then
               message = "Red for color 1 should be 0 not: " & r & vbCr
               End If
              If g <> 0 Then
               message = message & "Green for color 1 should be 0 not: " & g & vbCr
               End If
              If b <> 255 Then
               message = message & "Blue for color 1 should be 255 not: " & b & vbCr
            End If
                   CheckColor = message
            Exit Function

' 2 is: (r = 0, g = 255, b = 0)
        Case 2
                If r <> 0 Then
               message = "Red for color 2 should be 0 not: " & r & vbCr
               End If
              If g <> 255 Then
               message = message & "Green for color 2 should be 255 not: " & g & vbCr
               End If
              If b <> 0 Then
               message = message & "Blue for color 2 should be 0 not: " & b & vbCr
            End If
                  CheckColor = message
            Exit Function

' 3 is:  (r = 255, g = 0, b = 0)
        Case 3
                If r <> 255 Then
               message = "Red for color 3 should be 255 not: " & r & vbCr
               End If
              If g <> 0 Then
               message = message & "Green for color 3 should be 0 not: " & g & vbCr
               End If
              If b <> 0 Then
               message = message & "Blue for color 3 should be 0 not: " & b & vbCr
            End If
                 CheckColor = message
            Exit Function

' 4 is:  (r = 255, g = 255, b = 0)
        Case 4
                If r <> 255 Then
               message = "Red for color 4 should be 255 not: " & r & vbCr
               End If
              If g <> 255 Then
               message = message & "Green for color 4 should be 255 not: " & g & vbCr
               End If
              If b <> 0 Then
               message = message & "Blue for color 4 should be 0 not: " & b & vbCr
            End If
                  CheckColor = message
            Exit Function

' 5 is:  (r = 255, g = 0, b = 255)
        Case 5
                If r <> 255 Then
               message = "Red for color 5 should be 255 not: " & r & vbCr
               End If
              If g <> 0 Then
```

```
               message = message & "Green for color 5 should be 0 not: " & g & vbCr
               End If
              If b <> 255 Then
               message = message & "Blue for color 5 should be 255 not: " & b & vbCr
            End If
                CheckColor = message
            Exit Function

' 6 is: (r = 255, g = 127, b = 0)
        Case 6
                If r <> 255 Then
               message = "Red for color 6 should be 255 not: " & r & vbCr
               End If
              If g <> 127 Then
               message = message & "Green for color 6 should be 127 not: " & g & vbCr
               End If
              If b <> 0 Then
               message = message & "Blue for color 6 should be 0 not: " & b & vbCr
            End If
               CheckColor = message
            Exit Function

' 7 is:  (r = 0, g = 255, b = 255)
        Case 7
                If r <> 0 Then
               message = "Red for color 7 should be 0 not: " & r & vbCr
               End If
              If g <> 255 Then
               message = message & "Green for color 7 should be 255 not: " & g & vbCr
               End If
              If b <> 255 Then
               message = message & "Blue for color 7 should be 255 not: " & b & vbCr
            End If
               CheckColor = message
            Exit Function

' 8 is:  (r = 64, g = 64, b = 64)
        Case 8
                If r <> 64 Then
               message = "Red for color 8 should be 64 not: " & r & vbCr
               End If
              If g <> 64 Then
               message = message & "Green for color 8 should be 64 not: " & g & vbCr
               End If
              If b <> 64 Then
               message = message & "Blue for color 8 should be 64 not: " & b & vbCr
            End If
               CheckColor = message
            Exit Function

' 9 is:  (r = 192, g = 192, b = 192)
        Case 9
                If r <> 192 Then
               message = "Red for color 9 should be 192 not: " & r & vbCr
               End If
              If g <> 192 Then
               message = message & "Green for color 9 should be 192 not: " & g & vbCr
               End If
              If b <> 192 Then
               message = message & "Blue for color 9 should be 192 not: " & b & vbCr
            End If
              CheckColor = message
            Exit Function

' 10 is:  (r = 254, g = 0, b = 96)

        Case 10
                If r <> 254 Then
               message = "Red for color 10 should be 254 not: " & r & vbCr
               End If
              If g <> 0 Then
```

```
               message = message & "Green for color 10 should be 0 not: " & g & vbCr
               End If
              If b <> 96 Then
               message = message & "Blue for color 10 should be 96 not: " & b & vbCr
            End If
             CheckColor = message
            Exit Function
' 11 is:  (r = 160, g = 224, b = 0)
        Case 11
                If r <> 160 Then
               message = "Red for color 11 should be 160 not: " & r & vbCr
               End If
              If g <> 224 Then
               message = message & "Green for color 11 should be 224 not: " & g & vbCr
               End If
              If b <> 0 Then
               message = message & "Blue for color 11 should be 0 not: " & b & vbCr
            End If
              CheckColor = message
            Exit Function

' 12 is:  (r = 0, g = 254, b = 160)
        Case 12
                If r <> 0 Then
               message = "Red for color 12 should be 0 not: " & r & vbCr
               End If
              If g <> 254 Then
               message = message & "Green for color 12 should be 254 not: " & g & vbCr
               End If
              If b <> 160 Then
               message = message & "Blue for color 12 should be 160 not: " & b & vbCr
            End If
              CheckColor = message
            Exit Function

' 13 is:  (r = 128, g = 0, b = 160)
        Case 13
                If r <> 128 Then
               message = "Red for color 13 should be 128 not: " & r & vbCr
               End If
              If g <> 0 Then
               message = message & "Green for color 13 should be 0 not: " & g & vbCr
               End If
              If b <> 160 Then
               message = message & "Blue for color 13 should be 160 not: " & b & vbCr
            End If
              CheckColor = message
            Exit Function

' 14 is:  (r = 176, g = 176, b = 176)
        Case 14
                If r <> 176 Then
               message = "Red for color 14 should be 176 not: " & r & vbCr
               End If
              If g <> 176 Then
               message = message & "Green for color 14 should be 176 not: " & g & vbCr
               End If
              If b <> 176 Then
               message = message & "Blue for color 14 should be 176 not: " & b & vbCr
            End If
                 CheckColor = message
            Exit Function

' 15 is:  (r = 0, g = 240, b = 240)
        Case 15
                If r <> 0 Then
               message = "Red for color 15 should be 0 not: " & r & vbCr
               End If
              If g <> 240 Then
               message = message & "Green for color 15 should be 240 not: " & g & vbCr
               End If
              If b <> 240 Then
               message = message & "Blue for color 15 should be 240 not: " & b & vbCr
            End If
              CheckColor = message
            Exit Function

' 16 is:  (r = 240, g = 240, b = 240)
        Case 16
              If r <> 240 Then
               message = "Red for color 16 should be 240 not: " & r & vbCr
               End If
              If g <> 240 Then
               message = message & "Green for color 16 should be 240 not: " & g & vbCr
               End If
              If b <> 240 Then
               message = message & "Blue for color 16 should be 240 not: " & b & vbCr
            End If
              CheckColor = message
            Exit Function

' 17 is:  (r = 0, g = 0, b = 240)
        Case 17
                If r <> 0 Then
               message = "Red for color 17 should be 0 not: " & r & vbCr
               End If
              If g <> 0 Then
               message = message & "Green for color 17 should be 0 not: " & g & vbCr
               End If
              If b <> 240 Then
               message = message & "Blue for color 17 should be 240 not: " & b & vbCr
            End If
              CheckColor = message
            Exit Function


' 18 is:  (r = 0, g = 240, b = 0)
        Case 18
                If r <> 0 Then
               message = "Red for color 18 should be 0 not: " & r & vbCr
               End If
              If g <> 240 Then
               message = message & "Green for color 18 should be 240 not: " & g & vbCr
               End If
              If b <> 0 Then
               message = message & "Blue for color 18 should be 0 not: " & b & vbCr
            End If
             CheckColor = message
            Exit Function


' 19 is:  (r = 240, g = 0, b = 0)
        Case 19
                If r <> 240 Then
               message = "Red for color 19 should be 240 not: " & r & vbCr
               End If
              If g <> 0 Then
               message = message & "Green for color 19 should be 0 not: " & g & vbCr
               End If
              If b <> 0 Then
               message = message & "Blue for color 19 should be 0 not: " & b & vbCr
            End If
              CheckColor = message
            Exit Function

' 20 is:  (r = 240, g = 240, b = 0)

        Case 20
                If r <> 240 Then
               message = "Red for color 20 should be 240 not: " & r & vbCr
               End If
              If g <> 240 Then
```

```
               message = message & "Green for color 20 should be 240 not: " & g & vbCr
               End If
              If b <> 0 Then
               message = message & "Blue for color 20 should be 0 not: " & b & vbCr
            End If
             CheckColor = message
            Exit Function

' 21 is:  (r = 240, g = 0, b = 240)
        Case 21
                If r <> 240 Then
               message = "Red for color 21 should be 240 not: " & r & vbCr
               End If
              If g <> 0 Then
               message = message & "Green for color 21 should be 0 not: " & g & vbCr
               End If
              If b <> 240 Then
               message = message & "Blue for color 21 should be 240 not: " & b & vbCr
            End If
             CheckColor = message
            Exit Function

' 22 is:  (r = 240, g = 122, b = 0)
        Case 22
                If r <> 240 Then
               message = "Red for color 22 should be 240 not: " & r & vbCr
               End If
              If g <> 122 Then
               message = message & "Green for color 22 should be 122 not: " & g & vbCr
               End If
              If b <> 0 Then
               message = message & "Blue for color 22 should be 0 not: " & b & vbCr
            End If
             CheckColor = message
            Exit Function

' 35 is: r = 225, g = 0, b = 0)
        Case 35
                If r <> 225 Then
               message = "Red for color 35 should be 225 not: " & r & vbCr
               End If
              If g <> 0 Then
               message = message & "Green for color 35 should be 0 not: " & g & vbCr
               End If
              If b <> 0 Then
               message = message & "Blue for color 35 should be 0 not: " & b & vbCr
            End If
             CheckColor = message
            Exit Function

        Case Else
              CheckColor = ""
        End Select

End Function
```

```
Sub test_flipcase()
  Dim UserIn As String, Num As Long
  UserIn = InputBox(Prompt:="Enter some text:", _
                    Title:="FlipCase Test")
   Num = 5
   MsgBox flipcase(UserIn, Num)
   MsgBox UserIn
   MsgBox flipcase(UserIn)
   MsgBox UserIn

End Sub

Function flipcase(tStr As String, Optional nChar) As String
  Dim k As Long

  Dim testC As String * 1  'length of string as 1
   If IsMissing(nChar) Then
     nChar = Len(tStr)
   End If

  For k = 1 To nChar
  testC = Mid(tStr, k, 1)
   If (StrComp(testC, "A", vbBinaryCompare) >= 0) And _
       (StrComp(testC, "Z", vbBinaryCompare) <= 0) Then
       Mid(tStr, k, 1) = UCase(testC)
   ElseIf (StrComp(testC, "a", vbBinaryCompare) >= 0) And _
          (StrComp(testC, "z", vbBinaryCompare) <= 0) Then
       Mid(tStr, k, 1) = UCase(testC)
   End If
   Next k
   flipcase = tStr
End Function

Sub zoomExtents()
   Dim oView As View
   Dim pnt(1 To 4) As Point3d
   Dim intview As Integer
   intview = 1

   CadInputQueue.SendCommand "FIT VIEW EXTENDED 1"
   Set oView = ActiveDesignFile.Views(intview)

   pnt(1).x = oView.Origin.x
   pnt(1).Y = oView.Origin.Y
   pnt(2).x = oView.Extents.x
   pnt(2).Y = oView.Origin.Y
   pnt(3).x = oView.Extents.x
   pnt(3).Y = oView.Extents.Y
   pnt(4).x = oView.Origin.x
   pnt(4).Y = oView.Extents.Y

   pnt1 = oView.Extents
   oView.Redraw
   ActiveDesignFile.Fence.DefineFromModelPoints 1, pnt()
   ActiveDesignFile.Fence.Draw msdDrawingModeHilite

End Sub

Sub TestGetFiles()
   ' Call to test GetFiles function.
   Dim PPath As String
   Dim BellPath As String
   Dim dctDict As Dictionary
   Dim varItem As Variant
   Dim GetTempDir As String
   'GetTempDir = "C:\Users\knowles_keith\Desktop\Microstation_test_Folder"
   PPath = "P:\Active Projects\PGE\Substation\"
   BellPath = "6446 BELL\2000 Substation\2300 Engineering\2310 Electrical\2311 Drawings\Indoor"
   GetTempDir = PPath & BellPath
   'Create new dictionary.
   Set dctDict = New Dictionary
   ' Call recursively, return files into Dictionary object.
   If GetFiles(GetTempDir, dctDict, False) Then
      ' Print items in dictionary.
      For Each varItem In dctDict
         Debug.Print varItem
      Next
   End If
End Sub
Function GetFiles(strPath As String, _
                dctDict As Dictionary, _
                Optional blnRecursive As Boolean) As Boolean

   ' This procedure returns all the files in a directory into
   ' a Dictionary object. If called recursively, it also returns
   ' all files in subfolders.

   Dim fsoSysObj      As FileSystemObject
   Dim fdrFolder      As Folder
   Dim fdrSubFolder   As Folder
   Dim filFile        As File

   ' Return new FileSystemObject.
   Set fsoSysObj = New FileSystemObject

   On Error Resume Next
   ' Get folder.
   Set fdrFolder = fsoSysObj.GetFolder(strPath)
   If Err <> 0 Then
      ' Incorrect path.
      GetFiles = False
      GoTo GetFiles_End
   End If
   On Error GoTo 0

   ' Loop through Files collection, adding to dictionary.
   For Each filFile In fdrFolder.Files
      dctDict.Add filFile.path, filFile.path
   Next filFile

   ' If Recursive flag is true, call recursively.
   If blnRecursive Then
      For Each fdrSubFolder In fdrFolder.SubFolders
         GetFiles fdrSubFolder.path, dctDict, True
      Next fdrSubFolder
   End If

   ' Return True if no error occurred.
   GetFiles = True

GetFiles_End:
   Exit Function
End Function


Sub KJK()
 Dim ob As Application


  Application.ActiveDesignFile.TotalEditingTime


End Sub

Sub testScanFilter()
  Dim rng As Range3d
  Dim pnt3D As Point3d

  Dim mycell As CellInformation
  Dim myCellEnum As CellInformationEnumerator

  Dim myElem As Element
  Dim myEnum As ElementEnumerator
  Dim myFilter As New ElementScanCriteria
  Dim ElementCounter As Long
  Dim myCollection As New Collection
  'myFilter.ExcludeAllTypes
  myFilter.ExcludeAllLevels
  'myFilter.ExcludeAllColors
  'myFilter.IncludeType msdElementTypeText
  'myFilter.IncludeType msdElementTypeTextNode

  myFilter.IncludeLevel ActiveDesignFile.Levels("Border-titleblock")
  myFilter.IncludeLevel ActiveDesignFile.Levels("Border and Titleblock")
  'myFilter.IncludeLevel ActiveDesignFile.Levels("Level 1")
  'myFilter.IncludeLevel ActiveDesignFile.Levels("Existing")

  'myFilter.IncludeOnlyCell "BDR-D10"
  'myFilter.IncludeColor 4
  Set myEnum = ActiveModelReference.Scan(myFilter)

  While myEnum.MoveNext
    ElementCounter = ElementCounter + 1
    Set myElem = myEnum.Current
    myCollection.Add myElem
    MsgBox myElem.AsCellElement.Name & " " & "origin: " & vbLf & _
                  myElem.AsCellElement.Origin.x & ", " & myElem.AsCellElement.Origin.Y

    'MsgBox myElem.AsCellElement.Origin.x & " " & myElem.AsCellElement.Origin.Y
    'MsgBox myElem.AsCellElement.IsGraphical
    rng = myElem.AsCellElement.Range

    'MsgBox "x: " & Str(pnt3D.x = rng.High.x)
    pnt3D.x = rng.High.x
    pnt3D.Y = rng.High.Y
    MsgBox "High X: " & Str(pnt3D.x) & "High Y: " & Str(pnt3D.Y)
     pnt3D.x = rng.Low.x
    pnt3D.Y = rng.Low.Y
    MsgBox "Low X: " & Str(pnt3D.x) & "Low Y: " & Str(pnt3D.Y)
  Wend
  MsgBox ElementCounter & " elements found."
End Sub
```

```
Sub getRange()
 On Error GoTo errhnd
 Dim lngDspPrty As Long

 Dim ele As CellElement
 Dim success As Boolean
 success = False
 Dim rng As Range3d
 Dim BorderName As String

 Set ele = F2_G01.GetBorder(False)
 rng = ele.Range
 BorderName = ele.Name

 success = ZoomToTitle(rng, BorderName, 1)

 MsgBox success
errhnd:
   Select Case Err.number
     Case 91 'Get Border didn't find any Borders
             'Could be a raster file a raster Title Block
       MsgBox "Program ended! No title block on this drawing."
       Err.Clear
     End Select
End Sub

Function ZoomToTitle(Rngr As Range3d, BDR_X1X As String, viewNmbr As Integer) As Boolean
 Dim dblFactor As Double
 Dim DeltaY As Double
 Dim DeltaX As Double
 Dim oView As View
 Set oView = ActiveDesignFile.Views(viewNmbr)
 Dim pntOrigin As Point3d
 Dim rngExtents As Range3d
 Dim pntExtents As Point3d
 Dim myLine As LineElement
 Dim pntZoom As Point3d

 'Establish extents just around the Title Block area
 'This allows for extra elements outside of the Title Block
 'area to not affect the zoom into the title area of the Title
 'Block
 '*******************************************
 rngExtents = Rngr

 oView.Origin = rngExtents.Low

 pntExtents.x = rngExtents.High.x - rngExtents.Low.x
 pntExtents.Y = rngExtents.High.Y - rngExtents.Low.Y

 oView.Extents = pntExtents
 'oView.Redraw
 'oView.Redraw
 '********************************************

 pntOrigin.x = Rngr.Low.x
 pntOrigin.Y = Rngr.Low.Y
 pntOrigin.Z = 0

 DeltaX = Rngr.High.x - Rngr.Low.x
 DeltaY = Rngr.High.Y - Rngr.Low.Y

 Select Case BDR_X1X
    Case "BDR-D10"
        With Rngr
          pntZoom.x = .Low.x + (1.488623 * DeltaY)
          pntZoom.Y = .Low.Y + (0.2227318 * DeltaY)
        End With
        dblFactor = 0.43
     Case "BDR-E10"
```

```
        With Rngr
          pntZoom.x = .Low.x + (1.358326 * DeltaY)
          pntZoom.Y = .Low.Y + (0.163336 * DeltaY)
        End With
        dblFactor = 0.32
     Case "BDR-D12"
        With Rngr
          pntZoom.x = .Low.x + (1.488623 * DeltaY)
          pntZoom.Y = .Low.Y + (0.2227318 * DeltaY)
        End With
        dblFactor = 0.43
     Case "BDR-E12"
        With Rngr
          pntZoom.x = .Low.x + (1.358323 * DeltaY)
          pntZoom.Y = .Low.Y + (0.163336 * DeltaY)
        End With
        dblFactor = 0.32
     Case "BDR-T10"
        With Rngr
          pntZoom.x = .Low.x + (0.9702261904 * DeltaX)
          pntZoom.Y = .Low.Y + (0.16335 * DeltaY)
        End With
        dblFactor = 0.32
     Case "BDR-T12"
        With Rngr
          pntZoom.x = .Low.x + (0.9702214 * DeltaX)
          pntZoom.Y = .Low.Y + (0.16333 * DeltaY)
        End With
        dblFactor = 0.32
     Case Else1
       ZoomToTitle = False
       Exit Function
 End Select


      With Application
        Set myLine = .CreateLineElement2(Nothing, pntOrigin, pntZoom)
        .ActiveModelReference.AddElement myLine
      End With
 'Zoom about the center of the range.
 oView.ZoomAboutPoint pntZoom, dblFactor
 oView.Redraw
 oView.Redraw
 ZoomToTitle = True


End Function
```

```
Dim WithEvents myMS As Application

Private Sub Class_Initialize()

End Sub

Private Sub myMS_OnDesignFileOpened(ByVal DesignFileName As String)

Buttons


End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Attach Cell Library" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd J:\PGE\Cad Standards\PGE Cell Libraries\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd PGE_PikeDrafterCells.CEL"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Attach Cell Library

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Preferences [irasb]" Then

    '   Set a variable associated with a dialog box
        SetCExpressionValue "savePrefs.textEditorStyle", 4, "USERPREF"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Preferences [irasb]

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Preferences [irasb]" Then

    '   Set a variable associated with a dialog box
        SetCExpressionValue "savePrefs.textEditorStyle", 0, "USERPREF"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Preferences [irasb]

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Import Levels" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd J:\PGE\Cad Standards\PGE Cell Libraries\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd Electrical.levels.dgn"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Import Levels

    If DialogBoxName = "Level/Filter Import" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Level/Filter Import

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Attach Cell Library" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd J:\PGE\Cad Standards\PGE Cell Libraries\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd PGESHEET(new).CEL"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Attach Cell Library

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Attach Cell Library" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd J:\PGE\Cad Standards\PGE Cell Libraries\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd PGE_PikeDrafterCells.CEL"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Attach Cell Library

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Preferences [descartes]" Then

    '   Set a variable associated with a dialog box
        SetCExpressionValue "savePrefs.textEditorStyle", 0, "USERPREF"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Preferences [descartes]

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Attach Cell Library" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd J:\PGE\Cad Standards\PGE Cell Libraries\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd PGE_PikeDrafterCells.CEL"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Attach Cell Library

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Preferences [descartes]" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Preferences [descartes]

    If DialogBoxName = "Open" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd P:\Active Projects\PGE\Substation\6454 Rivergate II\2000 Substation\2300 Engineering\2310 Electrical\2311 Drawings\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd rvgt6603c1.dgn"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Open

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Preferences [descartes]" Then

    '   Set a variable associated with a dialog box
        SetCExpressionValue "savePrefs.textEditorStyle", 4, "USERPREF"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Preferences [descartes]

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Print - Raster Options" Then

    '   Set a variable associated with a dialog box
        SetCExpressionValue "rasterOptionsUI.quality", 100, "PLOTDLG"
    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Print - Raster Options

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Attach Cell Library" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd J:\PGE\Cad Standards\PGE Cell Libraries\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd PGE_PikeDrafterCells.CEL"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Attach Cell Library

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Color Table" Then

        CadInputQueue.SendCommand "CT= "

    End If  ' Color Table

    If DialogBoxName = "Design File Settings" Then

    '   The following statement opens modal dialog "Advanced Unit Settings"

    '   The following statement opens modal dialog "Alert"

    '   Set a variable associated with a dialog box
    '   This only modifies a few bits of the variable it changes. It first
    '   creates a mask for clearing the bits it will change. Then it gets
    '   the variable and uses the mask to clear those bits. Finally
    '   it sets the desired bits in the value and saves the updated value.
        lngTemp = Not 3
        lngTemp = GetCExpressionValue("dgnSet.unitFormatDGN", "DGNSET") And lngTemp
        SetCExpressionValue "dgnSet.unitFormatDGN", lngTemp Or 1, "DGNSET"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Design File Settings

    If DialogBoxName = "Alert" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Alert

    If DialogBoxName = "Advanced Unit Settings" Then

        SetCExpressionValue "dgnSet.adv_uorPerStorage", 10000, "DGNSET"
    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Advanced Unit Settings

    If DialogBoxName = "Import Levels" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd J:\PGE\Cad Standards\PGE Cell Libraries\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd Electrical.levels.dgn"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Import Levels

    If DialogBoxName = "Level/Filter Import" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Level/Filter Import

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Design File Settings" Then

    '   The following statement opens modal dialog "Advanced Unit Settings"

    '   The following statement opens modal dialog "Alert"

    '   Set a variable associated with a dialog box
    '   This only modifies a few bits of the variable it changes. It first
    '   creates a mask for clearing the bits it will change. Then it gets
    '   the variable and uses the mask to clear those bits. Finally
    '   it sets the desired bits in the value and saves the updated value.
        lngTemp = Not 3
        lngTemp = GetCExpressionValue("dgnSet.unitFormatDGN", "DGNSET") And lngTemp
        SetCExpressionValue "dgnSet.unitFormatDGN", lngTemp Or 1, "DGNSET"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Design File Settings

    If DialogBoxName = "Alert" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Alert

    If DialogBoxName = "Advanced Unit Settings" Then

        SetCExpressionValue "dgnSet.adv_uorPerStorage", 10000, "DGNSET"
    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Advanced Unit Settings

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Attach Cell Library" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd J:\PGE\Cad Standards\PGE Cell Libraries\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd PGESCH.CEL"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Attach Cell Library

    If DialogBoxName = "Attach Cell Library" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd J:\PGE\Cad Standards\PGE Cell Libraries\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd PGESHEET(new).CEL"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Attach Cell Library

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Color Table" Then

        CadInputQueue.SendCommand "CT= "

    End If  ' Color Table

    If DialogBoxName = "Color Table" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultCancel

    End If  ' Color Table

    If DialogBoxName = "Design File Settings" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Design File Settings

    If DialogBoxName = "Import Levels" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd J:\PGE\Cad Standards\PGE Cell Libraries\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd Electrical.levels.dgn"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Import Levels

    If DialogBoxName = "Level/Filter Import" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Level/Filter Import

    If DialogBoxName = "Open" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultCancel

    End If  ' Open

    If DialogBoxName = "Open" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd P:\Active Projects\PGE\Substation\6446 BELL\2000 Substation\2300 Engineering\2310 Electrical\2311 Drawings\Indoor\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd bell7313b0.dgn"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Open

    If DialogBoxName = "Design File Settings" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultCancel

    End If  ' Design File Settings

    If DialogBoxName = "Color Table" Then
    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultCancel

    End If  ' Color Table

    If DialogBoxName = "Design File Settings" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Design File Settings

    If DialogBoxName = "Design File Settings" Then

    '   The following statement opens modal dialog "Advanced Unit Settings"

    '   The following statement opens modal dialog "Alert"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Design File Settings

    If DialogBoxName = "Alert" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Alert

    If DialogBoxName = "Advanced Unit Settings" Then

        SetCExpressionValue "dgnSet.adv_uorPerStorage", 10000, "DGNSET"
    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Advanced Unit Settings

    If DialogBoxName = "Alert" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Alert

    If DialogBoxName = "Color Table" Then

        CadInputQueue.SendCommand "CT= "

    End If  ' Color Table

    If DialogBoxName = "Design File Settings" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Design File Settings

    If DialogBoxName = "Color Table" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultCancel

    End If  ' Color Table

    If DialogBoxName = "Import Levels" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd J:\PGE\Cad Standards\PGE Cell Libraries\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd Electrical.levels.dgn"
    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Import Levels

    If DialogBoxName = "Level/Filter Import" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Level/Filter Import

    If DialogBoxName = "Compress Options" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Compress Options

    If DialogBoxName = "Import Levels" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd J:\PGE\Cad Standards\PGE Cell Libraries\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd Electrical.levels.dgn"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Import Levels

    If DialogBoxName = "Level/Filter Import" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Level/Filter Import

    If DialogBoxName = "Open" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd P:\Active Projects\PGE\Substation\6446 BELL\2000 Substation\2300 Engineering\2310 Electrical\2311 Drawings\Indoor\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd bell7313c0.dgn"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Open

    If DialogBoxName = "Color Table" Then

        CadInputQueue.SendCommand "CT= "

    End If  ' Color Table

    If DialogBoxName = "Color Table" Then

    End If  ' Color Table

    If DialogBoxName = "Import Levels" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd J:\PGE\Cad Standards\PGE Cell Libraries\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd Electrical.levels.dgn"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Import Levels

    If DialogBoxName = "Level/Filter Import" Then
    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Level/Filter Import

    If DialogBoxName = "Design File Settings" Then

    '   The following statement opens modal dialog "Advanced Unit Settings"

    '   The following statement opens modal dialog "Alert"

    '   This only modifies a few bits of the variable it changes. It first
    '   creates a mask for clearing the bits it will change. Then it gets
    '   the variable and uses the mask to clear those bits. Finally
    '   it sets the desired bits in the value and saves the updated value.
        lngTemp = Not 3
        lngTemp = GetCExpressionValue("dgnSet.unitFormatDGN", "DGNSET") And lngTemp
        SetCExpressionValue "dgnSet.unitFormatDGN", lngTemp Or 1, "DGNSET"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Design File Settings

    If DialogBoxName = "Alert" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Alert

    If DialogBoxName = "Advanced Unit Settings" Then

        SetCExpressionValue "dgnSet.adv_uorPerStorage", 10000, "DGNSET"
    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Advanced Unit Settings

    If DialogBoxName = "Design File Settings" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Design File Settings

    If DialogBoxName = "Design File Settings" Then

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Design File Settings

    If DialogBoxName = "Color Table" Then

    End If  ' Color Table

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Attach Cell Library" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd J:\PGE\Cad Standards\PGE Cell Libraries\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd PGESCH.CEL"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Attach Cell Library

    If DialogBoxName = "Attach Cell Library" Then

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setDirectoryCmd J:\PGE\Cad Standards\PGE Cell Libraries\"

        CadInputQueue.SendCommand "MDL COMMAND MGDSHOOK,fileList_setFileNameCmd PGESHEET(new).CEL"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Attach Cell Library

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Print Attributes" Then

    '   Set a variable associated with a dialog box
        SetCExpressionValue "plotAttrUI.line_wghts", 0, "PLOTDLG"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Print Attributes

End Sub
```

```
Implements IModalDialogEvents
Private Sub IModalDialogEvents_OnDialogClosed(ByVal DialogBoxName As String, ByVal DialogResult As MsdDialogBoxResult)

End Sub

Private Sub IModalDialogEvents_OnDialogOpened(ByVal DialogBoxName As String, DialogResult As MsdDialogBoxResult)

    If DialogBoxName = "Print Attributes" Then

    '   Set a variable associated with a dialog box
        SetCExpressionValue "plotAttrUI.line_wghts", 0, "PLOTDLG"

    '   Remove the following line to let the user close the dialog box.
        DialogResult = msdDialogBoxResultOK

    End If  ' Print Attributes

End Sub
```