

## Easy PDF

```
Sub Main()  
    Dim oDoc As Document  
    Dim oPDFAddIn As TranslatorAddIn  
    Try  
        Dim ThisApp = ThisApplication  
        Dim TransObj As TransientObjects = ThisApp.TransientObjects  
        oDoc = ThisDoc.Document  
        If oDoc.DocumentType <> 12292 Then '12292 = kDrawingDocument object  
            MessageBox.Show("This Rule must be run from within a Drawing  
Document.", "Incorrect Document Type")  
            Exit Sub  
        End If  
        oPDFAddIn = ThisApp.ApplicationAddIns.ItemById("{0AC6FD96-2F4D-42CE-  
8BE0-8AEA580399E4}")  
  
        oContext = TransObj.CreateTranslationContext  
        oContext.Type = IOMechanismEnum.kFileBrowseIOMechanism  
        oOptions = TransObj.CreateNameValueMap  
        oDataMedium = TransObj.CreateDataMedium  
  
        If oPDFAddIn.HasSaveCopyAsOptions(oDataMedium, oContext, oOptions)  
Then  
            oOptions.Value("All_Color_AS_Black") = 0 ' 0 = False, 1 = True  
            oOptions.Value("Remove_Line_Weights") = 1 ' 0 = False, 1 = True  
            oOptions.Value("Vector_Resolution") = 4800 ' DPI  
            oOptions.Value("Sheet_Range") =  
Inventor.PrintRangeEnum.kPrintAllSheets  
            'oOptions.Value("Custom_Begin_Sheet") = 2  
            'oOptions.Value("Custom_End_Sheet") = 4  
            End If  
  
            'Set the PDF target file name  
            oDataMedium.FileName = ThisDoc.Path & "\" & ThisDoc.FileName(False) &  
".pdf"  
            Catch  
                MessageBox.Show("Error Getting Document Information" & vbCr & "Is the  
current Document Saved?", "Document Error")  
            End Try  
            'Publish document  
            Try  
                oPDFAddIn.SaveCopyAs(oDoc, oContext, oOptions, oDataMedium)  
            Catch  
                MessageBox.Show("Unable to Save PDF" & vbCr & "Is the file open or  
read only?", "Error Saving Document")  
            End Try  
        End Sub
```

## Easy iProperties

```
PartNumber = "Sample_2"  
Description = "iLogic for Anyone: Sample 2"  
ProjectName = "Sample 2"  
ProjectNo = "121516_01"  
  
'Standard iProperties  
  
iProperties.Value("Project", "Part Number") = PartNumber  
iProperties.Value("Project", "Description") = Description  
  
'custom iProperties  
iProperties.Value("Custom", "PROJECT NAME") = ProjectName  
iProperties.Value("Custom", "PROJECT NUMBER") = ProjectNo  
  
iLogicVb.UpdateWhenDone = True
```

## Better Save

```
'Set a reference to the document  
Dim oDoc As Document  
oDoc = ThisApplication.ActiveDocument  
  
'Set a reference to the top node of the active browser  
Dim oTopNode As BrowserNode  
oTopNode = oDoc.BrowserPanes.ActivePane.TopNode  
Dim oNode As BrowserNode  
  
For Each oNode In oTopNode.BrowserNodes  
    ' If the node is visible and expanded, collapse it.  
    If oNode.Visible = True And oNode.Expanded = True Then  
        oNode.Expanded = False  
    End If  
Next  
  
'Return view to Home view  
ThisApplication.CommandManager.ControlDefinitions.Item _  
("AppViewCubeHomeCmd").Execute  
  
'zoom all  
ThisApplication.ActiveView.Fit
```

## Rapid Drawings

```
Sub Main()  
  
    Try  
        If ThisDoc.Document.DocumentType <> 12292 Then '12292 =  
kDrawingDocument object  
            MessageBox.Show("This Rule must be run from within the Drawing  
Document.", "Incorrect Document Type")  
            Exit Sub  
        End If  
        oldDrawing = ThisDoc.Document  
  
        InventorVb.Application.CreateFileDialog(oFileDialog)  
        With oFileDialog  
            .DialogTitle = "Select the File to Create a Drawing of"  
            .Filter = "Autodesk Inventor Components (*.ipt;  
*.iam)|*.ipt;*.iam"  
            '.SuppressResolutionWarnings = True  
            '.OptionsEnabled = True  
            .MultiSelectEnabled = False  
            '.CancelError = True  
            .ShowOpen()  
        End With  
        sFile = oFileDialog.FileName  
  
        If sFile = "" Then Exit Sub  
  
        idwFile = Left(sFile, Len(sFile) - 3) & ".idw"  
        oldDrawing.SaveAs(idwFile, True)  
        newDrawing = ThisApplication.Documents.Open(idwFile, True)  
  
        Dim oFD As FileDescriptor  
        oFD =  
newDrawing.ReferencedFileDescriptors(1).DocumentDescriptor.ReferencedFileDesc  
riptor  
        oFD.ReplaceReference(sFile)  
        newDrawing.Update()  
        newDrawing.Save  
  
        Catch ex As Exception  
            newDrawing.ReleaseReference  
            MessageBox.Show("Error on " & sFile & ":" & vbCrLf & ex.message)  
        End Try  
  
End Sub
```

\*These code samples are derived from and include content available publicly online provided by others  
\*MESA makes no warranty of the provided samples and you acknowledge you use them at your own risk  
\*These samples are provided "as-is" and with no assurances that they are free of defects or errors