If Not TypeOf ThisApplication.ActiveDocument Is PartDocument Then

MsgBox("A Part document must be 'active' for this rule to work. Exiting rule.",,"")

Exit Sub

End If

Dim oPDoc As PartDocument = ThisApplication.ActiveDocument

If Not TypeOf oPDoc.ComponentDefinition Is SheetMetalComponentDefinition Then

MsgBox("This Part is not a Sheet Metal Part. Exiting rule.",,"")

Exit Sub

End If

Dim oSMDef As SheetMetalComponentDefinition = oPDoc.ComponentDefinition

If Not oSMDef.HasFlatPattern Then

oAns = MsgBox("This Sheet Metal Part does not have a Flat Pattern." & vbCrLf & \_

"This is needed to calculate 'LaserCutLength'." & vbCrLf & \_

"Do you want this rule to automatically create the Flat Pattern, then continue?", vbYesNo + vbQuestion, "")

If oAns = vbNo Then Exit Sub

Try

oSMDef.Unfold

Catch oEx As Exception

MsgBox("Failed to 'Unfold' this Sheet Metal Part, to create a Flat Pattern." & vbCrLf & \_

oEx.Message & vbCrLf & oEx.StackTrace, , "")

Exit Sub

End Try

End If

Dim oFP As FlatPattern = oSMDef.FlatPattern

Dim oLengths As New List(Of Double)

For Each oEL As EdgeLoop In oFP.BottomFace.EdgeLoops

oLengths.Add(ThisApplication.MeasureTools.GetLoopLength(oEL))

Next

a = InputListBox("", oLengths, 0.0, "Loop Lengths", "Individual Loop Lengths")

Dim oParam As UserParameter

Try

oParam = oSMDef.Parameters.UserParameters.Item("LaserCutLength")

oParam.Value = oLengths.Sum

Catch

oParam = oSMDef.Parameters.UserParameters.AddByValue("LaserCutLength", oLengths.Sum, "mm")

End Try