Sub InsertInletIntoAssembly()

' This code places the inlet nozzle into our assembly

' The inlet nozzle consists of a pipe and an end connection (i.e. flange, capped flange, or valve)

' The user has the option to place the inlet nozzle on the top of the tank, or the front dish head plate

' If they place it on the dish head plate, it must be located near the top of the tank, and not the bottom half

Dim strInletTubeName, strFlangeFile As String

' No new geometry is created for inlets - they only use existing parts from the library

' This let's us find the right name of the tube (or pipe) based on the inlet size

strInletTubeName = LIBRARY\_PATH & "Flanges\ANSI B36.10 XS - " & INLET\_SIZE & ".ipt"

' This code uses our "GetFlangeFilename" function to find the name of the end connection based on

' flange type, flange end connection, and inlet size

strFlangeFile = GetFlangeFilename(INLET\_FL\_TYPE, INLET\_FL\_END, INLET\_SIZE)

' Define the matrices that will be needed to place the inlet nozzle, including the pipe and flange

Dim matrixM, matrixN As DocumentUnitsMatrix

Dim strInletPipeBrowserName, strInletFlangeBrowserName As String

' If the user wants to place the inlet nozzle on the top, use these locating matrices

If INLET\_LOC = "Top" Then

' These strings will be used to set the occurrence names in the browser to indicate they are installed on top of the tank

strInletPipeBrowserName = "Top Inlet Pipe - " & INLET\_SIZE & " Inch:1"

strInletFlangeBrowserName = "Top Inlet Flange - " & INLET\_SIZE & " Inch:1"

' This matrix represents the orientation required for the pipe on top of the tank

matrixM = ThisDoc.Geometry.Matrix(1, 0, 0, 0, 0, 0, -1, TANK\_OD / 2 + 6, 0, 1, 0, TANK\_L / 2 - INLET\_OFF, 0, 0, 0, 1)

' The locating matrix will be different for open, capped and valve end connection choices

If INLET\_FL\_END = "Open" Then

' If the user chooses a welding neck flange, a different offset matrix value will be required for the Y (up) direction

If INLET\_FL\_TYPE = "Welding Neck" Then

matrixN = ThisDoc.Geometry.Matrix(0, 1, 0, 0, -1, 0, 0, TANK\_OD / 2 + dblFlangeOffsetDistance + 9 in, 0, 0, 1, TANK\_L / 2 - INLET\_OFF, 0, 0, 0, 1)

Else

matrixN = ThisDoc.Geometry.Matrix(0, 1, 0, 0, -1, 0, 0, TANK\_OD / 2 + dblFlangeOffsetDistance + 6 in, 0, 0, 1, TANK\_L / 2 - INLET\_OFF, 0, 0, 0, 1)

End If

ElseIf INLET\_FL\_END = "Capped" Then

matrixN = ThisDoc.Geometry.Matrix(1, 0, 0, 0, 0, 0, 1, TANK\_OD / 2 + 6, 0, -1, 0, TANK\_L / 2 - INLET\_OFF, 0, 0, 0, 1)

Else

matrixN = ThisDoc.Geometry.Matrix(1, 0, 0, 0, 0, 0, 1, TANK\_OD / 2 + dblFlangeOffsetDistance + 7 in, 0, -1, 0, TANK\_L / 2 - INLET\_OFF, 0, 0, 0, 1)

End If

' If the user wants to place the inlet nozzle on the front, this is the code that will be used to create the location matrices

Else

' These strings will be used to set the occurrence names in the browser to indicate they are installed on top of the tank

strInletPipeBrowserName = "Front Inlet Pipe - " & INLET\_SIZE & " Inch:1"

strInletFlangeBrowserName = "Front Inlet Flange" & INLET\_SIZE & " Inch:1"

Dim dblDishOffset As Double = TANK\_L / 2 + (INLET\_OFF / (TANK\_OD / 2)) \* DISH\_DEPTH + 6

matrixM = ThisDoc.Geometry.Matrix(-1, 0, 0, 0, 0, 1, 0, TANK\_OD / 2 - INLET\_OFF, 0, 0, -1, dblDishOffset + 4, 0, 0, 0, 1)

' Since we created a function (GetFrontOrRearMatrix) that figures out location matrices on the front and rear dish plates,

' we can take advantage of that and don't need to figure them out separately, like we had to for the top

matrixN = GetFrontOrRearMatrix(INLET\_FL\_TYPE, INLET\_FL\_END, INLET\_OFF, 10, INLET\_SIZE, "Front", "Top")

End If

' These are the iLogic commands to add the pipe and flange components to the assembly, and place them properly based on the matrices

Dim componentM = Components.Add(strInletPipeBrowserName, strInletTubeName, position := matrixM, grounded := True, visible := True, appearance := Nothing)

Dim componentN = Components.Add(strInletFlangeBrowserName, strFlangeFile, position := matrixN, grounded := True, visible := True, appearance := Nothing)

End Sub