Function GetFrontOrRearMatrix(strFlangeType As String, strFlangeEnd As String, dblVertOffset As Double, dblCustomHorOffset As Double, \_

dblFlangeSize As Double, strSide As String, strTopOrBottom As String) As DocumentUnitsMatrix

' This function returns a matrix object that is derived based on all of its inputs

' It is only good for matrices on the front dish head plate, and the rear dish head plate, and only for end connections

' That includes flanges, caps and valves

Dim matrixReturn As DocumentUnitsMatrix

' This variable calculates the length from the center of the tank to the outside edge of the tank body

' It then approximates the dish head plate depth using a linear formula (which isn't always the most accurate)

' The goal is to get the distance as from tank centerline to the outside edge of the tank, including the dish head plate

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

' This uses the "GetFlangeOffsetDistance" function to get the initial offset values based on the type of end connection

Dim dblFlangeOffset As Double = GetFlangeOffsetDistance(strFlangeType, strFlangeEnd, dblFlangeSize)

Dim dblYValue, dblZValue As Double

' We need to know if the end connection will be on the upper half of the tank, or the lower half of the tank

' If it's on the upper half, our Y location value will be positive

' If it's on the lower half, our Y location value will be negative

If strTopOrBottom = "Top" Then

dblYValue = TANK\_OD / 2 - dblVertOffset

Else

dblYValue = -TANK\_OD / 2 + dblVertOffset

End If

' For "Open" end connections, calculate our Z location value, and create one matrix for the front, and one for the rear

' The reason front and rear placement matrices differ, is that a flange has to be rotated 180-degrees if it's placed

' on the rear dish head; in other words, you always want the flanges pointing away from the tanks

If strFlangeEnd = "Open" Then

dblZValue = dblDishOffset + dblFlangeOffset + dblCustomHorOffset - 6 in

If strSide = "Front" Then

matrixReturn = ThisDoc.Geometry.Matrix(0, 0, 1, 0, 0, 1, 0, dblYValue, -1, 0, 0, dblZValue, 0, 0, 0, 1)

Else

matrixReturn = ThisDoc.Geometry.Matrix(0, 0, -1, 0, 0, 1, 0, dblYValue, 1, 0, 0, -dblZValue, 0, 0, 0, 1)

End If

' For "Capped" end connections, calculate our Z location value, and create one matrix for the front, and one for the rear

ElseIf strFlangeEnd = "Capped" Then

dblZValue = dblDishOffset + dblCustomHorOffset

If strSide = "Front" Then

matrixReturn = ThisDoc.Geometry.Matrix(1, 0, 0, 0, 0, 1, 0, dblYValue, 0, 0, 1, dblZValue, 0, 0, 0, 1)

Else

matrixReturn = ThisDoc.Geometry.Matrix(-1, 0, 0, 0, 0, 1, 0, dblYValue, 0, 0, -1, -dblZValue, 0, 0, 0, 1)

End If

' For "Valve" end connections, calculate our Z location value, and create one matrix for the front, and one for the rear

Else

dblZValue = dblDishOffset + dblCustomHorOffset + 1 in

If strSide = "Front" Then

matrixReturn = ThisDoc.Geometry.Matrix(1, 0, 0, 0, 0, 1, 0, dblYValue, 0, 0, 1, dblZValue, 0, 0, 0, 1)

Else

matrixReturn = ThisDoc.Geometry.Matrix(-1, 0, 0, 0, 0, 1, 0, dblYValue, 0, 0, -1, -dblZValue, 0, 0, 0, 1)

End If

End If

' Set our resulting matrix to the GetFrontOrRearMatrix function so that it can be returned to our calling statement

GetFrontOrRearMatrix = matrixReturn

End Function