



Manufacturing Innovation. Together.

```
Sub CreateAndConfigureTankBody()  
    ' This code creates a new copy of the tank body assembly template in our new folder structure  
    ' It then updates the tank body geometry based on values we pass to the assembly  
    Dim strNewTankBodyFileName As String  
    ' This string represents the new name of our unique, copied tank body assembly file  
    strNewTankBodyFileName = PROJECT_PATH & PROJECT_ID & "\Tank Body Assy\tank Body Assy - " & PROJECT_ID & ".iam"  
  
    ' We first check to make sure the tank body assembly file has not been previously created  
    If System.IO.File.Exists(strNewTankBodyFileName) = False Then  
        ' This here is a sample of how to make a variable that represents an assembly document  
        ' It uses the Inventor API, which you can use freely (for the most part) throughout iLogic rules  
        Dim subAssyl As AssemblyDocument  
        ' This statement tells subAssyl to represent the template file, and opens it up in the Inventor interface  
        subAssyl = ThisApplication.Documents.Open(TEMPLATE_PATH & "Tank Body Assy\tank Body Assy.iam", True)  
        ' This code is like selecting "File Save As" in the Inventor interface, and we now have our new file saved  
        subAssyl.SaveAs(strNewTankBodyFileName, False)  
        ' This can be taken from an iLogic snippet, and is used to insert components into assemblies  
        ' This code inserts our newly created tank body assembly into our master tank assembly file  
        Dim componentA = Components.Add("Tank Body Assy:1", strNewTankBodyFileName, position := Nothing, grounded := True, visible := True, appearance := Nothing)  
        ' This will now close the new tank body assembly file  
        subAssyl.Close  
        ' This code calculates the horizontal and vertical locations of our gunline assembly (for later use)  
        GUNLINE_HOR_OFF = Round((TANK_OD / 2) * .6667)  
        GUNLINE_VERT_OFF = Round((TANK_OD / 2) * .25)  
        ' This code represents our typical manway offset, which is 26" above the bottom of the tank  
        MANWAY_VERT_OFF = -TANK_OD / 2 + 26  
        ' These statements pass parameters from our master assembly file into the tank body assembly file  
        Parameter("Tank Body Assy:1", "PROJECT_ID") = PROJECT_ID  
        Parameter("Tank Body Assy:1", "PROJECT_PATH") = PROJECT_PATH  
        Parameter("Tank Body Assy:1", "TANK_OD") = TANK_OD  
        Parameter("Tank Body Assy:1", "TANK_L") = TANK_L  
        Parameter("Tank Body Assy:1", "SHELL_W_1") = SHELL_W_1  
        Parameter("Tank Body Assy:1", "SHELL_W_2") = SHELL_W_2  
        Parameter("Tank Body Assy:1", "SHELL_Q_1") = SHELL_Q_1  
        Parameter("Tank Body Assy:1", "SHELL_Q_2") = SHELL_Q_2  
        Parameter("Tank Body Assy:1", "GUNLINE_SIZE") = GUNLINE_SIZE  
        Parameter("Tank Body Assy:1", "GUNLINE_VERT_OFF") = GUNLINE_VERT_OFF  
        Parameter("Tank Body Assy:1", "GUNLINE_HOR_OFF") = GUNLINE_HOR_OFF  
        Parameter("Tank Body Assy:1", "MANWAY_VERT_OFF") = MANWAY_VERT_OFF  
        Parameter("Tank Body Assy:1", "GUNLINE") = GUNLINE  
        Parameter("Tank Body Assy:1", "MANWAY") = MANWAY  
        Parameter("Tank Body Assy:1", "F_HATCH") = F_HATCH  
        Parameter("Tank Body Assy:1", "F_HATCH_OFF") = F_HATCH_OFF  
        Parameter("Tank Body Assy:1", "R_HATCH") = R_HATCH  
        Parameter("Tank Body Assy:1", "R_HATCH_OFF") = R_HATCH_OFF  
        Parameter("Tank Body Assy:1", "INLET_OFF") = INLET_OFF  
        Parameter("Tank Body Assy:1", "INLET_PIPE_OD") = INLET_PIPE_OD  
        ' Once all the parameters are updated in the tank body assembly file, we want to run their rules  
        ' This will allow the tank body assembly to update all its own parts and components itself  
        iLogicVb.RunRule("Tank Body Assy:1", "Size Dish")  
        iLogicVb.RunRule("Tank Body Assy:1", "Calculate and Place Shells")  
    End If  
End Sub
```