' This part of the rule pushes the relevant parameters to the straight pipe and angled pipe

' so that they update properly

Parameter("Sump-Straight Pipe:1", "SUMP\_PIPE\_PROJ") = SUMP\_PIPE\_PROJ

Parameter("Sump-Straight Pipe:1", "SUMP\_SIZE") = SUMP\_SIZE

Parameter("Sump-Straight Pipe:1", "SUMP\_H") = SUMP\_H

Parameter("Sump-Straight Pipe:1", "SUMP\_FLOOR\_OFF") = SUMP\_FLOOR\_OFF

Parameter("Sump-Straight Pipe:1", "TANK\_OD") = TANK\_OD

Parameter("Sump-Straight Pipe:1", "DISH\_DEPTH") = DISH\_DEPTH

Parameter("Sump-Angled Pipe:1", "SUMP\_SIZE") = SUMP\_SIZE

Parameter("Sump-Angled Pipe:1", "SUMP\_H") = SUMP\_H

Parameter("Sump-Angled Pipe:1", "SUMP\_FLOOR\_OFF") = SUMP\_FLOOR\_OFF

' The default template is made with 6" bracket

' If the user selects an 8" sump, then we need to replace the bracket

' Try...Catch allows us to try operations and catch errors if they occur

' In this case, we just want the automation to keep going if it fails for some reason to find the right bracket

' Without "Try...Catch" functionality, the code would come to a complete stop and exit the whole process

' if there was some error replacing the bracket

If SUMP\_SIZE = 8 in Then

Try

Component.Replace("Sump Bracket - 6:1", LIBRARY\_PATH & "Hardware\Sump Bracket - 8.ipt", True)

Catch

' Do nothing if it can't find the right part in the model browser

End Try

Else

Try

Component.Replace("Sump Bracket - 8:1", LIBRARY\_PATH & "Hardware\Sump Bracket - 6.ipt", True)

Catch

' Do nothing if it can't find the right part in the model browser

End Try

End If