• How is (are) the tool(s) made available to users? As a .tbx file with one or more .py files? As stand-alone .py files?

The tools are made available to users as a .tbx file (Find Adjacent & Neighboring Polygons.tbx) and contains two Python scripts (one for each tool).

• What is the folder structure, if any, of the files that make up the tool?

In order to use the tools, the files have to be unzipped.

• What type of documentation is provided, separate from the script itself? Is there a Help page on the tool dialog box?

Each tool had item descriptions in ArcCatalog plus many in-script comments.

Open the .py files in PythonWin and review the script’s documentation.

See if you can recognize the following elements of a script:

• Importing modules, such as ArcPy and others

import arcpy

import sys

import traceback

• Receiving parameters from the tool dialog box, such as

GetParameterAsText

inputFC = arcpy.GetParameterAsText(0)

idfieldName = arcpy.GetParameterAsText(1)

newField = arcpy.GetParameterAsText(2)

newField2 = arcpy.GetParameterAsText(3)

• Error-handling techniques, such as the try-except statement

try:

sWksp = arcpy.env.scratchWorkspace = "in\_memory"

arcpy.env.overwriteOutput = True

# Get the input feature layer…..

except:

tb = sys.exc\_info()[2]

tbinfo = traceback.format\_tb(tb)[0]

pymsg = "PYTHON ERRORS:\nTraceback Info:\n" + tbinfo + "\nError Info:\n " + \

str(sys.exc\_type)+ ": " + str(sys.exc\_value) + "\n"arcpy.AddError(pymsg)

• Custom functions, such as def

None Exist

• Message handling, such as the AddMessage and AddWarning

Functions

msgs = "GP ERRORS:\n" + arcpy.GetMessages(2) + "\n"arcpy.AddError(msgs)