\_\_ProjectCreator\_\_.py – main file, has main loop, GUI creation and manipulation, and links between other files via buttons.

callback.py – checks variables brought in by \_\_projectCreator\_\_, decides what files to send to based off user input.

directoryFinder.py – searches highlight database job numbers to see if users’ job has been created and has a directory.

entryMan.py – manipulates entry variables for automatic directory finder.

toBlankAccess.py – demo access process to send data to empty access file.

autoDrawingsEditLayers.py – new style for auto machines, turning on and off layers.

drawingsEdit.py – old style for semi auto machines, using scripts only.

drawingsEditLayer.py – new style for semi auto machines, turning on and off layers.

exExcelMan.py – exports and manipulates data from pages 29, 30, 32 from semi auto machines to csv files.

exportAccessDrawings.py – converts csv data to access database.

exportExcelDrawings.py – converts csv data to excel database.

printAutoDrawings.py – print drawings, auto style.

printDrawings.py – print drawings, semi auto style.

Acad.doc.SendCommand() = sends command to autocad from python program

Timer code:

Layer example:

timer = time.perf\_counter()

while True:

  endTimer = time.perf\_counter()

   if (endTimer - timer) >= 60: return

   try:

**acad.doc.SendCommand('-LAYER F SYN2 \n\n')**

      break

   except: pass

script example:

timer = time.perf\_counter()

while True:

endTimer = time.perf\_counter()

if (endTimer - timer) >= 60: return

try:

**acad.doc.SendCommand('SCRIPT ' + scriptDir + '\\29-noColdOtherScript.scr \n')**

    break

  except: pass