# -\*- coding: utf-8 -\*-  
import pandas as pd  
import asyncio  
import glob  
  
# Шаги: 1)Скопировал файл в другой, разделив все данные в равных частях на три листа  
# 2) Создал три цикла, для записи данных в три других файла, для равномерного распределения данных  
# 3) Подписал столбы, где адреса {Address} и прохожусь по ним  
# 4) Ускорение и объединение файлов...  
# 5) Время выполнения 3:30 - 3:40 ... Слишком долго  
dff = pd.read\_excel('C:\\Users\\11.xlsx', sheet\_name="List1")  
dss = pd.read\_excel("C:\\Users\\11.xlsx", sheet\_name="List2")  
dtt = pd.read\_excel("C:\\Users\\11.xlsx", sheet\_name="List3")  
  
  
async def lst1(dff):  
 dal = dff['Address'].value\_counts()[dff['Address'].value\_counts() > 1]  
 await asyncio.sleep(0.1)  
 dal.to\_excel("C:\\Users\\Ya1\\1more.xlsx")  
  
  
async def lst2(dss):  
 dlh = dss['Address'].value\_counts()[dss['Address'].value\_counts() > 1]  
 await asyncio.sleep(0.1)  
 dlh.to\_excel("C:\\Users\\Ya1\\2more.xlsx")  
  
  
async def lst3(dtt):  
 dla = dtt['Address'].value\_counts()[dtt['Address'].value\_counts() > 1]  
 await asyncio.sleep(0.1)  
 dla.to\_excel("C:\\Users\\Ya1\\3more.xlsx")  
  
  
ioloop = asyncio.get\_event\_loop()  
tasks = [  
 ioloop.create\_task(lst1(dff)),  
 ioloop.create\_task(lst2(dss)),  
 ioloop.create\_task(lst3(dtt))  
]  
ioloop.run\_until\_complete(asyncio.wait(tasks))  
ioloop.close()  
  
all\_data = pd.DataFrame()  
for f in glob.glob('C:/Users/Ya1/\*.xlsx'):  
 df = pd.read\_excel(f)  
 all\_data = all\_data.append(df, ignore\_index=True)  
  
writer = pd.ExcelWriter('C:/Users/Ya1/coll.xlsx', engine='xlsxwriter')  
all\_data.to\_excel(writer, sheet\_name='Sheet1')  
writer.save()

Программа заработает, только если подписать в файле название столбцов, ну или хотя-бы подписать {Address} над адресами объектов. Я сместил перенесенные данные из главного файла в новый файл со сдвигом на 1 вниз, чтобы подписать столбец {Address}.

Итог: Записал в файл адреса объектов, на которых продажи > 1. Файл “coll” содержит в себе три файла меньшего размера, где посчитано кол-во повторений адресов.