1. Move raw files from GoAnywhere to J:\Acctng\Production\Data using Python

* Go into GoAnywhere folders of each carrier. Manually copy and paste into Production\data directory
* File Type for each carrier:

MHS: .txt

JH: .txt

Symetra: .xlsx(Needed to be aggregate to one full monthly spreadsheet)

MBen: .xlsx

PRU: .xlsx

MAG: .xlsx

Guardian: .xlsx

Pacific Life: .xlsx

Nationwide: .xlsx

Principal: .xlsx

Lincoln: .xlsx

MetLife: .xlsx

ERA: .xlsx

UNUM: .xlsx

Penn Mutual: .xlsx (needed to be aggregated)

Master Care: .xlsx once quarterly

Protective: .txt file (Needed to be aggregated into one full .txt)

Voya: .xlsx

1. Formatting

* Currently using Excel for this step. Import data from the text files or excel file, and extract columns that we want. Export as csv file and waited to be loaded.
* Aggregate the Excel files or text files if needed in Python
* Format the file to .csv in Python

Reasons: Each carrier file comes in in a different format. Python has a lot more flexibility to deal with these data files. (Pandas, Openpyxl, pywin32)

1. Load csv file into PremiumStaging Table by BCP

* Currently using SSIS package for this step to load the carriers csv file into PremiumStaging
* After csv files are available, copy csv files into C:\dev\Production\data\YYYY\MM
* use bcp to import these files into PremiumStaging
* Some carriers are year-to-date, and the other are not.
* For those who are not, retrieve last month data from PremiumDetail, copy into premiumStaging and modify ‘SourceFileName’, ‘Row ID’, ‘ImportDate’, ‘YearApplied’, ‘MonthApplied’.

1. Data Mapping

* There are missing IDs or new producer name occasionally. Currently look up and fill manually by searching product table and producer table.
* I think it is harder to automate this step as it involves some judging. In most cases, writing out a stored procedure might work.
  + Use Producer\_Mapping Table

1. Run Stage to Live Proc

* Currently doing step 4 and 5 by executing scripts in <month>ProdandOverrides.sql. There are some data cleaning and updating procedures required before pushing data from PremiumStaging to PremiumDetail, such as padding 0’s in front of JH policy numbers.
* Can move some of the scripts and stored proc into instruction.json
* Final Push procedure contains several stored Proc. This can be automated by adding the stored procs into instruction.json.

**\* steps for how to reset data if things go wrong when pushing from staging to detail table.**

1. Special Request:

Can automate the following process by Python.(Submitting query and sending out emails to different carriers)

* Pacific Life:
* **Tami Wroblewski** [twroblewski@pacificlife.com](mailto:twroblewski@pacificlife.com)

PL Production thru <Month> <Year>.xlsx

PL Market Share YTD <Month> <Year>.xlsx

LTC Production - <Month> <Year>.xlsx

* Kati Anderson Kati.Anderson@mfin.com

J:\Acctng\Financials\STORY\2022\Production 2022.xlsx

J:\ Acctng\Production\Production Process Documentation\SQL Queries\Production Story.sql

Format File: J:\Acctng\Production\2022\Requests\Kati Anderson\

* Derek Palumbo [DPalumbo@jhancock.com](mailto:DPalumbo@jhancock.com)

\\mfh.local\data\CorpData\Acctng\Production\2022\Requests\Derek Palumbo

This is needed **quarterly**

* Thomas Financial
  + **Leslie Darrigo** ldarrigo@thomasfinancial.com

[Sales by Member Firm Report](http://mreportsdev.mfin.com/Reports/Pages/Report.aspx?ItemPath=%2fSales+Reports%2fSales+By+MemberFirm&ViewMode=Detail)

202206 Sales By MemberFirm.xlsx

1. Download the Excel file and save to [Request](file:///J:\Acctng\Production\2022\Requests\Leslie%20Darrigo) folder
2. Unmerge columns A to B
3. Move Confidential heading and Report title to column C
4. Remove column B
5. Merge and Center B1:J1 and B2:J2
6. Auto fit column A
7. Add filters for B19:K19
8. Same step for sheet 2

Steps for resetting loop:

MHS Data

1. Push P-1999-LIF-1.txt into MHSDetail table with bcp
2. Run spPMHSUpdateDate and spPMHSInsertData to push data into MHSPremiumStaging
3. Run spPMSANNMapIDs for missing IDs
4. if there is missing IDs, process terminated. Insert statements will be added to the script C:\dev\Production\etl\2022\07\1.0 NewMonthMappings.sql, which will then be added to instructions.json
5. Reset steps create a copy of **dbo.Mapping** and **dbo.Product** , saved the copy tables as **dbo.Mapping\_temp** and **dbo.Product\_temp**

Non MHS Data

1. Push non-MHS .txt files into PremiumDetail with bcp
2. Run[dbo].[PopulatePremiumStaging\_M\_IDs] to check missing IDs. Ifyes, added to insert statements and instructions.json
3. if there is missing IDs, process terminated. Insert statements will be added to the script C:\dev\Production\etl\2022\07\1.0 NewMonthMappings.sql, which will then be added to instructions.json
4. Reset steps create a copy of **dbo.Mapping** and **dbo.Product** , saved the copy tables as **dbo.Mapping\_temp** and **dbo.Product\_temp**

Reset:

1. Up to this point, tables that got manipulated are: **dbo.MHSDetail**, **dbo.MHSPremiumDetail**, **dbo.PremiumStaging**, **dbo.Mapping,** and possibly, **dbo.Product**
2. delete from MHSPremiumDetail and PremiumDetail
3. Reload non MHS txt files and rerun spPMHSInsertData to push MHS data into MHSPremiumStagin
4. Copy data from product table and mapping table from their corresponding temp table
5. Insert value into mapping and product tables.
6. Run C:\dev\Production\etl\2022\07\1.1 MHSJHPolicy.sql to update JH policy number
7. Run Populatepremiumstaging\_JH to update JH policy numbers
8. If something going wrong, go back to step 2 of Reset.