**ED Rx Webscraping Design Notes**

Working from the inside out, core code is now working for retrieving the data needed. The purpose of this document is to define the overall design from the outside in.

Even though this project is limited to analyzing the price of ED specific meds across a limited group of zipcodes, it makes sense to design this as a webscraping script that can take *any* list of Rx meds along w/*any* list of zipcodes to produce an output file w/the same information. This way, a clean data file can be created for any group of meds against any group of zipcodes for future analysis projects.

***Input Files:***

These files need to be placed into the data directory prior to running the program:

***rx.csv***

Each line contains the following csv fields:

* Name
* Dose
* Number of Pills

***zipcode.csv***

One line of csv fields

***Output Files:***

These files will be written to the data directory as the program is running:

***results.csv***

Each line contains the following csv fields:

* Rx Name (comes directly from input)
* Rx Dose (comes directly from input)
* Rx Number of Pills (comes directly from input)
* Zipcode (comes directly from input)
* Pharmacy Name (no cleaning needed)
* Pharmacy List Price (strip text, dollar sign, and comma to output as integer)
* Pharmacy Discount Price (strip text, dollar sign, and comma to output as integer)

This is a clean file of the webscraped data that can now be read in for data analysis purposes.

***log.txt***

This file will be written to after the following successful actions:

* Rx.csv has been read in successfully.
* Zipcode.csv has been read in successfully.
* Program start time is ‘timestamp’
* X Number of records for Rx and Zipcode have been successfully written to results.csv w/timestamp (Repeated)
* All input files have been processed successfully and the resulting output file should contain X number of records.
* Program end time is ‘timestamp’

The purpose of this file is to:

* Have easy visibility of all steps that occurred successfully or unsuccessfully in running the script.
* Provide performance information (timestamp data) to show how long it took to run the script based on the parameters passed in the input files.
* Provide information as to how many records should be in the results.csv file so it can be compared to how many records are actually in the file (for QA purposes)

**Outer Logic:**

* Initialize results.csv and log.txt
* Read in rx.csv and zipcode.csv
* Create list of URLS from data in rx.csv
* For i in URL list:
  + Open URL
  + For i in zipcode list:
    - Set zip code
    - Gather data
    - Check that #pharm = #list = #disc
    - Write to DF
  + Clean DF
  + Write to results.csv (append)
  + Write to log.csv (append)