16 HW #7B: Pandas Joins (II) [TBD]

NEEDS TO BE REWRITTEN AK

Repeat HW #7A, this time using Pandas. In order to receive full credit, please turn in a document which is python code containing what would be run to return the data asked.

The queries below rely on information from the stock return data. To load the data use the following commands. **Note: these load retdate as a date, rather than a string**

The best approach to learning from these problems is to complete them using pen and paper, working by yourself and then using your group to double check your results. The First Five problems provide a short overview of the core concepts in the assignment, so make sure that you understand them. The Main Problems section contains questions which range from easy to very difficult. Remember to don't get stuck! If a problem is taking a long time or is too difficult, *use your group!*

```
## Initial Information
import pandas as pd
import numpy as np

df2010D = pd.read_csv('/Users/ncross/git/sqlnotes/newserver/data/2010.tdf',
        sep='\t', engine='python', names=['symb', 'retdate', 'opn', 'high', 'low',
        'cls', 'vol', 'exch'], parse_dates=['retdate'])

df2011D = pd.read_csv('/Users/ncross/git/sqlnotes/newserver/data/2011.tdf',
        sep='\t', engine='python', names=['symb', 'retdate', 'opn', 'high', 'low',
        'cls', 'vol', 'exch'], parse_dates=['retdate'])

dffnd = pd.read_csv('/Users/ncross/git/sqlnotes/newserver/data/fnd.tdf',
        sep='\t', engine='python', names=['gvkey', 'datadate', 'fyear', 'indfmr',
        'consol', 'popsrc', 'datafmt' , 'tic', 'cusip', 'conm', 'fyr', 'cash', 'dp',
        'ebitda', 'emp', 'invt', 'netinc', 'ppent', 'rev' , 'ui', 'cik'])

dfMTA = pd.read_csv('../sql-data/raw_data/mta/MTA_Hourly.tdf', sep='\t',
        engine='python', names=['plaza', 'mtadt', 'hr', 'direction', 'vehiclesez',
        'vehiclescash'])

dfTrans = pd.read_csv('../sql-data/raw_data/soapData.tdf', sep='\t',
        engine='python', names = ['orderid', 'userid', 'trans', 'type', 'local',
        'trans_dt', 'units', 'coupon', 'months', 'amt' ])
```

First Five

1. Write a query which returns the userid, trans_dt, amt and the total amount that the user has spent up to (but not including) that *day's* transaction. Make sure that if there is no previous transaction that the amount is set to *zero* and isn't null.

2. Write a query which returns a purchase number for each order. In other words, for each row return the amount, userid, date and the number of the sale, incrementing from one for each order.

3. For each local in the table, return the most common month of an order.

4. For each local in the table, return the *second* most common month of an order.

5. Create a dataset which has the following information: (1) userid, (2) date of first transaction and (3) number of transactions within the first 35 days of their first transaction.

Main Problems

1. Using the data from 2010, write a query which returns the seven day moving average for each stock's closing price. Only look at stocks whose symbols begin with the letter "A". Note that this should *not* be the last seven points, but instead the last seven *days*, not including the current day.
2. For each stock from 2010, write a query which returns the symbol, the closing price, the return date and the closing price on the previous day it was traded. Note that you just want to take the price from the previous row, if the rows are ordered by return date. Also, only do this for stocks that begin with the letter 'A'.
3. For each symbol in 2010, return the day(s) where it has its highest volume traded[12]
4. Return the closing price on the first and last day that the stock is traded in 2010.
5. How many missing days are there total? Make sure to only count missing days **after** a symbol has been in the data. So if a stock doesn't appear in the data until February, January does not count as missing. If a stock leaves the market before the end of the year, you can either count the days past their exit as missing or as not missing, just be consistent across all stocks.[13]
6. For each symbol that appears in 2011, calculate the number of missing trading days that it has in January 2010.

[12] There could be multiple days for a symb.

[13] In other words, if a stock leaves the data it maybe because the stock delisted, in which case the data is not missing.