

CS 620–Introduction to Data Science and Analytics, HW3

You have just been hired as an analyst for an investment firm. Your first assignment is to analyze data for stocks in the S&P 500. The S&P 500 is a stock index that contains the 500 largest publicly traded companies.

You have been given two sources of data to work with. The first is an XML file that contains the Symbol (ticker), company name, sector, and industry for every stock in the S&P 500, as of summer 2016. The second is a CSV file that contains pricing information for stocks in the S&P 500 between August 2009 and August 2010. There is one row in the CSV file for every stock, on every date that the market was open. Each row contains the date as a string, the stock's ticker, the day's opening price, the day's high price, the day's low price, the day's closing price, and the volume traded that day.

Use the provided files SP500_ind.csv and SP500_symbols.xml and the [starter code](#). Implement using python and associated libraries.

- Complete the following functions in your python module

```
def ticker_find(xml_root, ticker):
    """This function takes in the root of the xml dataset and a
    Symbol (ticker). Return the name of the ticker
    Ex: for ticker "A", the function returns Agilent Technologies Inc
    """

def calc_avg_open(csv_data_sp500, ticker):
    """This function takes in the csv_data_sp500 and a ticker.
    Return the average opening price for the stock as a float.
    """

def calc_vwap(csv_data_sp500, ticker):
    """This function takes in the csv_data_sp500 and a ticker. Return the volume weighted
    average price (VWAP) of the stock as a float. In order to do this, first find the average
    price of the stock on each day. Then, multiply that price with the volume on that day.
    Take the sum of these values. Finally, divide that value by the sum of all the volumes.
    (hint: average price for each day = (high + low + close)/3)
    """

def calc_mfi(company_data):
    """
    Compute Money Flow Index for a given company.
    """
```

1. To perform these calculations, you should call above functions in a logical order, with the appropriate parameters. Use the given driver code. Warning: **DO NOT CHANGE** the driver code. You are free to experiment with the driver code, but don't forget to revert it back to the original form. It's advised to run the program during development for a single company (example ticker: "IBM") until you get the correct results.

Note that stocks move in and out of the S&P 500. Some stocks may be represented in the CSV file, but not in the XML file (and vice-versa). Display "No data in SP500" for the names of these tickers. For example, for ticker DYN,

Ticker: DYN Name: No data in SP500 Avg: 2.11 VWAP: 2.12 MFI: 46.62

2. Calculate the Money Flow Index for all the companies.
Money Flow Index (MFI) is an indicator that measures the selling and buying pressure by

analyzing price and volume data. The MFI indicator swings between 0 and 100. MFI is generally created for a certain duration for a ticker (e.g., 14 days), but in this problem, we are going to calculate MFI for all the given days for a certain ticker in the dataset.

First calculate `typical_price` for each day which is the average of high, low and close.

$$\text{typical_Price} = (\text{Low} + \text{High} + \text{Close}) / 3$$

Next calculate the money flow which is `typical_price` X volume of the day.

$$\text{money_flow} = \text{Volume} \times \text{typical_price}$$

Then, two lists of `postive_flow` and a `negative_flow` over all the given days are formed. `postive_flow` list contains `money_flow` values those days where the typical price is higher than the previous day's typical price. `negative_flow` list contains `money_flow` values those days where the typical price is lower than the previous day's typical price. **Note that you need to add the previous days money flow when you append values to the list.** Don't forget to add zero in each cases. If there's `positive_flow` then add zero to `negative_flow` and vice versa. Also add zero to both `postive_flow` and `negative_flow` when there's no change. (Hint: Use if-elif-else conditions)

Next, calculate the `money_ratio` given by,

$$\text{money_ratio} = \text{total of positive_flow} / \text{total of negative_flow}$$

Compute the Money Flow Index (MFI).

$$\text{MFI} = 100 - [100 / (1 + \text{money_ratio})]$$

3. You **CANNOT USE** any regular python loops inside `calc_avg_open` and `calc_vwap` functions. Instead use DataFrame techniques learned in Pandas. You are free to use any regular loops or other techniques inside `calc_mfi` function.

What to turn in:

Lastname-hw3.py should contain the following information at the top:

CS620

HW3

@author:

Submit your .py file to Blackboard.

Due: Sunday, March 14, 2021, 11.59pm