Python Exercises

Date, I/O and Exception

1. Go back to ex9_dictionary exercise 2.
    a. Add logic such that if the user write “save”, each line in the dictionary is saved to file called “saved_dict.txt”. if there is already a file called “saved_dict.txt” overwrite it with the new dictionary (using the open mode ‘w’)
    b. Add logic such that if the user write “load”, everything in the “saved_dict.txt” is loaded into the dictionary.
2. Go back to the code in ex_if_else_bool exercise 1 (with addon code from ex_functions exercise 3).
   Remove the if, that checks if the first and second number is numeric and use the try except instead.
   Also change the if logic for divide by zero (question 1c) and use a try except instead.
3. Go to Yahoo finance and pick a company of interest. Then go to the Historical Data tab, and set “Frequency” to Daily, and “Time Period”>1 years, apply and download a csv of the data.
    a. Start by importing the file, and loop through the first 15 lines to understand how the csv looks like and is built up.
    b. Make a function that takes one line as input, and output two values, the date (in python date format,

without time), and a dictionary using a format where each column name is keys, and the value for that column is value, example:
{‘open’: 610.23, ‘high’: 612.56, ‘low’: 601.52, …}

c. Loop through all lines in the CSV and create a large dictionary with all information. Use the date (in python date format) as key, and a dictionary of the stock values from that date as values. Use the function form b
d. Normally you would expect 7 values for each line in the CSV (Date,Open,High,Low,Close,Adj Close,Volume). You also expect each pris to be numeric and date to be a certain date format.
update you code such that it can handle bad csv lines. Instead of writhing if/else, for each problem (where there might be a bunch we haven’t thought of yet), use try/except.
Just try to handle the line, if it was bad and the try failed, print a message with the full line.
Test you code by creating a bad line and check that it works. And verify that the code keeps going beyond the bad line.
e. Using the dictionary find the most profitable day (biggest % difference in open and close prices)
f. Find the highest price ever, what day is it, and what was the price

g. Extra challenging,
   There is a hypothesis that if the stock price increase one day (close is higher than open), than there is a higher chance of increasing the next day too. Test this hypothesis on your data by comparing probability of a random day having a stock price increase or having a stock price increase the day after an increase.