ISGB 79AA - Advanced Python for Financial Programming
Assignment 1 – Reading Excel Spreadsheets with Python and Pandas

This assignment has two parts. The first part involves working with income statement data using pandas read_excel(). The second part involves downloading stock time series from Yahoo Finance, creating a spreadsheet, and reading this spreadsheet using xlrd.

You should submit 4 separate notebooks for this assignment (3 notebooks for part A, and 1 notebook for part B). Name notebooks like: LastnameFirstnameAsn1-<question>, for example, SmithJohnAsn1-A1. Include in each cell of your notebook a Python comment at the start of the cell, describing the cell's statements.

Part A – Income Statement Consolidation and Analysis

A company uses a standard spreadsheet template for each division to provide their income statement data. The income statement submitted by each division of the company includes the past 2 years of actual results, as well as a year of a forecasted plan. (See Appendix A for additional background.). The company needs to consolidate the division's income statements to see the overall company actual and forecasted income statement. The company also needs to analyze each division's plans using selected accounting ratios.

A.1 Write a Python notebook (named like LastnameFirstnameAsn1-A1), that reads the following spreadsheets, using pandas read_excel():
incstmt-0.xlsx – template for income statements (all 0 results)
incstmt-diva.xlsx – income statement for division A
incstmt-divb.xlsx – income statement for division B
and combines division A and division B results into a total company income statement. All spreadsheets have two historical columns (for actual years 2018 and 2019) and one future column (projected for 2020).
Notes:
- Use the spreadsheet's first column (named "CODE") as an index (row label) for the DataFrames
- The spreadsheets have comment lines beginning with a #
- The column labelled "Description" can be dropped from the combined income statement
- Consider using the DataFrame add() method to add DataFrames
- This notebook should assume there are only two divisions, each with their own .xlsx file. So for this notebook, you use hardcoded statements such as pd.read_excel('incstmt-diva.xlsx', …)

A.2 Using your notebook written for part A.1 as a reference point, create a new notebook (named like LastnameFirstnameAsn1-A2) that could easily handle any number of divisions, in order to create a composite income statement.
Notes:
- Create a list of division names whose spreadsheets are to be combined, e.g.,
  divisions = ['diva', 'divb']
- Use a for loop to read each division's .xlsx, add cumulate a composite company-wide income statement in a pandas DataFrame
- Use a DataFrame corresponding to the worksheet incstmt-0.xlsx as a starting point for cumulating income statements.

Part A – Income Statement Consolidation and Analysis (continued)

A.3 Write a separate notebook (named like LastnameFirstnameAsn1-A3) that analyzes each division's results. In particular, display two tables.

One table should display the historical sales growth (from 2018 to 2019) as well as the projected sales growth (from 2019 and 2020). A trend indicator (+ or -) should be included in the table, as to whether sales growth is declining or increasing. For example:

```
Sales Growth Review:
Div  2018-2019  2019-2020  trend
A      -8.2%      24.6%      +
B       3.9%       2.9%      -
```

The second table should display the operating margin (EBT / SALE), and indicate if the sales growth trend is improving or declining. For example:

```
Operating Margin Review:
Div  Act2019  Proj2020  trend
A      28.3%     39.0%    +
B      33.4%     32.2%    -
```

notes:

- As in part A2, write this notebook in a general way so that it could easily handle any number of divisions. In other words, create a list of divisions, and use a for loop to calculate and display the above tables.
- The Sales Growth Review table requires calculating growth between adjacent years.
- The Operating Margin Review table values are separate for each year. For example, in the above table, division A's operating margin for 2019 of 28.3% is based on its 2019 EBT / 2019 SALES.

Part B – Reading a Stock Price Spreadsheet, cell-by-cell

B.1 For a stock of interest, download a .csv of 1 year of daily price data from Yahoo Finance. See Appendix B for information about downloading Yahoo Finance data.
B.2 Use Excel to import your downloaded .csv from step B.1 into an Excel spreadsheet, and save this as a .xlsx
Important: Do not Save As using the File Format: "Strict Open XML Spreadsheet (.xlsx)". xlrd will not be able read this type of file.

B.3 Use xlrd to read the .xlsx data into a pandas DataFrame. Do not use read_excel(), instead use lower-level xlrd methods such as open_workbook, sheet_by_index, cell_value, etc.

B.4 Using the pandas DataFrame created in step B.3, calculate and append a column for daily returns.

B.5 Create a histogram for the daily returns calculated in step B.4, using matplotlib.

Appendix A – Income Statement Background

An income statement covers a company's (or division's) sales, costs, and expenses for a given period, to determine the profits or losses for that period. A period may be historical (past) or projected (future). Some key measures:

Gross Profit = Sales – Cost of Goods Sold

Total Expenses = sum of individual expenses, such as:
- SG&A – Selling, General, and Administrative
- Advertising
- Depreciation and Amortization
- Rental Expense
- Other Expenses

Earnings before Taxes = Gross Profit – Total Expense

Sales Growth: $(Sales_{t+1} – Sales_t) / Sales_t$

Operating Margin = Earnings before Taxes / Sales

Appendix B – Downloading Data from Yahoo Finance

1) In a web browser, go to finance.yahoo.com

2) Search for a company of interest, in the search box at top.

3) choose the Historical Data tab, in the middle of the page

4) select the time period of 1Y, then Done

5) select frequency of Daily

6) select Apply

7) Download data via Download Data. The downloaded file is a .csv.