**P1 – Excel Grade Summary System**

**Overview:**

**This is a group project**. Assume a high school teacher approached your group complaining about the excel file their grading system produces each quarter. It spits out all the classes they teach with all their students in a single spreadsheet, with student information stored in a single column. The teacher wants your group to make a program that will automatically format and summarize the important information about each of the classes they teach.

**Libraries Required:**

* import openpyxl
* from openpyxl import Workbook
* from openpyxl.styles import Font

**External Files Required:**

* see the Learning Suite project description for file downloads. These are examples of excel files that you want to import, take the data, and create new excel workbooks that are better organized. The columns are Class Name (what class the student/grade is for), Student Info (it shows the last name, first name, and studentID all in one column separated by an underscore), and Grade (a grade in the class from 0 to 100).

**Logical Flow:**

Using the openpyxl library, import one of the two example excel files. Your program should be robust enough to be able to work either of the files, but when first writing your program, just choose one to work with.

Your program should also create a new workbook object that you’ll eventually save as a new excel file (that way you’ll still have the original excel file and a new excel file at the end).

* When you create a new workbook object, it automatically creates a worksheet called “Sheet”. You can get rid of it by using something like outputWB.remove(outputWB["Sheet"])

Your program will need to:

1. Create new worksheets for each class (e.g., a sheet for Algebra, a sheet for Calculus, etc.)
2. In each sheet, create columns for last name, first name, student ID, and grade with the student data for that class placed there.
3. A filter should be placed over the 4 aforementioned columns in each sheet.
4. Additionally, each sheet should have some simple summary information about each class using functions in columns F (the titles) and G (the data). It should show:
   * The highest grade
   * The lowest grade
   * The mean grade
   * The median grade
   * The number of students in the class
5. Some simple formatting (bolding headers) and changing the width of the columns.
   * The width of the columns for A,B,C,D,F,G must each be set to the number of characters in the header + 5.
   * For example the column D header is “Grade” which has 5 characters, so the width of column D should be 10, etc.
6. Save the results as a new Excel file named “formatted\_grades.xlsx”

See the example output for what this all should look like

There is a lot of variation in how exactly your group could perform this, so there isn’t one specific “logical flow” of how to do it. All that matters is that you create a program that could take any excel file that is in the same format as the 2 starter excel files and output another excel file formatted like the example output. This project is actually great practice for situations like this where you know what you’re starting with and what the end product should be, but you have to plan out the process of getting from A to B.

However, here are some hints that might help you implement each requirement:

1. Sheets for each class:
   1. When creating the new sheets, you’ll use .create\_sheet().
   2. myWorkbook.sheetnames gives a list of all the sheet names. This might be useful to check if you already have a sheet created for a class. E.g. if “Algebra” not in myWorkbook.sheetnames, make another sheet, but if “Calculus” is already there, don’t make a new sheet for it, etc.
   3. Remember you can loop through rows of a worksheet using .iter\_rows(). If you do iter\_rows(min\_row=2) you’ll skip the headers.
2. last name, first name, student ID, and grade columns
   1. If you loop through the original excel file, during every iteration of the loop, you can add a row from the original file to the new workbook.
   2. first name, last name, and studentID are stored all in one column. Cleaning up poorly formatted data is a common task. I recommend looking up the .split() function in python to help with this.
   3. One easy way to add all this data to the new workbook/worksheet is to use the worksheetVariable.append() function. You can put a list in the parentheses, and it will just add it to the next open spot in the worksheet.
3. Filter
   1. See 19.9 in the textbook if you need an example of this.
   2. myWorkbook.worksheets gives a list of worksheet objects that you could loop through to do this for each worksheet
   3. You need to apply the filter to the range starting in A1 and ending in D(the max number of rows in that sheet). How do you get the number of rows that have data in a sheet?
4. Adding functions
   1. See 19.8 in the textbook for examples of inserting excel functions.
   2. Column F will have the titles of the functions
   3. Column G will have the actual results
5. Simple formatting
   1. You only need to bold the headers of Columns A, B, C, D, F, and G.
   2. You need to adjust the width of those same columns based on the number of characters in each of the headers. What function returns the number of letters in a string?
6. Save the results
   1. Use myWorkbook.save(filename=”filename.xlsx”)

Upload just the python file to Learning Suite. Only one person per group needs to upload.

**Example Output:**

See the example\_output.xlsx file in the Learning Suite assignment description