COMP 1516 – Programming Fundamentals with Python – Assignment 2

Instructor	Mike Mulder (mmulder10@bcit.ca or Slack)
Total Marks	18
Due Dates	Sunday, Nov. 12th at midnight

Overview

You will be continuing with your script and modules from Assignment 1. You will be adding file input/output and will update the existing reports.

You may continue with your submission from Assignment 1 (preferred) or use the sample Assignment 1 implementation on D2L. Use the same patterns from Assignment 1 for your implementation of Assignment 2.

Your assignment must have no syntax errors and must successfully run (i.e., no runtime errors) for all combinations of the above inputs and outputs AND against the provided unit test. **If the script cannot be run, you will receive a mark of zero on this assignment.**

Requirements

The following table identifies the changes required for this assignment.

Requirement	Marks
Update the data.py module so that the get_student_grades function gets its list of student grades from the given data.csv file: • The student data from data.csv should be read in the program as a list, where each line in the file is an element in the list. • Any extra line returns should be removed from the end of each element. • The returned list of strings should be identical in format to the previous implementation	2

Summary Report Updates
 	
The Summary Report will be updates with the list of schools (sorted alphabetically) and the average, minimum and maximum grades in each school:
 	
`> main.py summary`
`Report:  2023-06-14 10:40:47`
`Number Schools:  3`
`Schools:  BCIT,SFU,UBC`
	2

Number Courses: 6 Number Students: 120 **School Averages:** **BCIT: 77.2%** **SFU: 78.0%** **UBC: 77.8%** **School Minimums:** **BCIT: 20.2%** **SFU: 26.8%** **UBC: 36.2%** **School Maximums:** **BCIT: 99.3%** **SFU: 99.1%** **UBC: 94.5%**	
School Report Updates The School Report will be updated with the list of courses in the school (listed alphabetically) and the top and bottom student based on their individual average grades in their courses. > main.py school BCIT Report: 2023-06-14 10:45:06 School: BCIT Number Courses: 3 **Courses: COMM1000,COMP1516,MATH1305** Number Students: 40 Average Grade: 77.2% Median Grade: 85.8% **Top Student: A0001007** **Top Grade: 97.5%** **Bottom Student: A0001018** **Bottom Grade: 30.3%**	2
The Summary Report should be printed to the console AND written to a file named summary.txt	1
The School Report should be printed to the console AND written to a file named as follows: <lowercase school name>.txt So for a school name of BCIT the filename would be bcit.txt	1
The provided unit tests (test_main.py) must run and all pass to verify the above requirements. You may have to tweak the report output to get the unit tests to pass.	10

Make sure your program works both on the command line and with the unit tests (test_main.py). The majority of your marks will come from the fully passing unit tests.

Submission

Submit the following Python files in a **zipfile** called **assignment2.zip** to the Assignment 2 dropbox in D2L:

- main.py – Your implementation of the main script
- grade_stats.py – Your implementation of the statistic module
- reports.py – Your implementation of a reports module
- data.py – Your implementation of a data module

Make sure you followed naming best practices and included DocString in all functions.

Summary

Implementation As per the requirements described above.	18 marks
Marks will be subtracted for violations of naming standards or missing DocString. (-0.5 mark each)	
Total	**18 marks**

Best Practices

- Variable names should be lower_snake_case
- Function names should be lower_snake_case
- All functions should include a DocString comment