420-LCU-05 Programming in Python - Assignment 2
Due Oct 09th , 2018 at 11:59 p.m.

1- **Identification section**: This section must be either in a comment, with a '#' preceding each line, or enclosed within triple quotes ('''). The grader and I need this section for the accurate processing of your assignment. Assignments missing this may lose up to 5% of the mark.

"""
Your Name and ID
420-LCU Computer Programming, Section #
S. Hilal, instructor
Assignment 2
"""

2- **Submission**: Submit your assignment in 1 Python file, with the extension .py. No need to create a ZIP file. Be sure to respect other instructions specified in the assignment. An important part of each assignment is to correctly follow the instructions closely.

3- **Late assignments** are accepted up to 1 week from deadline. But late penalty will be applied.

For this assignment, you will develop a small application that will enable a teacher to **enter**, **analyze** and **report** on the grades of the students in one class based on their marks in different components of the course. It can calculate a student's **overall grade**, **letter grade**, and the **class average**. Your program can define any functions that you find useful to perform some of the required operations.

Learning Objectives:

- Practice using lists, strings and loops.
- Multidimensional data structures
- built-in functions and methods for strings and lists
- User-defined functions (optional)

Description of Data:

A student can be identified by a name (first name only) and 3-digit ID. For each student, there are 6 grades based on 2 tests (20% each) and 4 assignments (15% each). Your program should be able to process the information for any number of students but you can limit your tests to 10 students.

Your program will define a 2-dimensional list (list of lists). **students** = []

The list **students** will hold the data for all students. Each list element holds the data for 1 student. Here is an example for 3 students:

Students = [
      ["Anne",234, 20, 18, 12, 10, 15, 11],
      ["Bob",123, 15, 18, 12, 15, 15, 12],
      ["Ben",124, 15, 18, 12, 15, 13, 12]]

Each list contains a student's name (a string), ID number (an integer), two test scores (out of 20) and four assignment scores (out of 15). All numeric data is stored as int.

Running Your Program:

The program starts by presenting the user with a "menu" of options. The user selects an option by typing the corresponding digit. When the program completes the processing of a given option, **the menu will be displayed again** and the user prompted to make another selection. This continues until the user selects exit.
Your program must do the required tests to make sure that it does the valid options only. (E.g. cannot process option 2 if students list is empty). **Define other validation** tests and add appropriate messages. Make sure to document validation tests in your comments.

The program menu

Welcome to the Teacher's Simple Class Calculator. Here's the list of options:

1- Enter student records (Name, ID, and 6 marks separated by commas)
2- Display the class average.
3- Display the total grade, letter grade and relation to class average for a given student
4- Display a simple bar chart to show grade distribution.
5- Exit

Select an option by entering its number or 5 to exit:

Description of the Different Options (above):

1- **Option 1**: The program will keep asking the user to enter a student record until the user enters "done". **Computer output in bold**. User input shown in red
    Example:
    Enter Student Record (Separate by commas, no spaces) or done: Anne,234,20,18,12,10,15,11
    Record Accepted
    Enter Student Record (Separate by commas, no spaces) or done: Bob,124,15,18,12,15,15,12
    Record Accepted
    Enter Student Record (Separate by commas, no spaces) or done: Ben,222,15,18,12,15

Record incomplete. Record rejected.

Enter Student Record (Separate by commas, no spaces) or done: Ben,222,15,18,12,15,13,12

Record Accepted

Enter Student Record (Separate by commas, no spaces) or done: Bob,235,20,18,12,15,15,12

Record Accepted

Enter Student Record (Separate by commas, no spaces) or done: John,222,20,19,12,15,13,12

Duplicate ID number. Record rejected

Enter Student Record (Separate by commas, no spaces) or done: done

- There may be duplicate names but IDs are unique.
- The program rejects an incomplete record.
- A complete record is stored in the list **students**.

2- **Option 2**: The class average for the entered grades will be displayed.

   E.g. Class Average = 84

3- **Option 3**: The program will ask the user to enter the name and ID of the student. The program will print the numeric grade, letter grade and deviation from average.

   Enter the name and ID of the student: Anne,234

   Grade for Anne ID = 234: 86 B, 2 points above average.

4- **Option 4**: **We will work on this option in the lab**. Not required for Assignment 2.

Below is a table of the letter grades that correspond to the total score:

Total Grade	Letter Grade
87 or above	A
From 75 to 86 inclusive	B
From 65 to 74 inclusive	C
Below 65	F

Writing your program:

1- You can use any of the list/strings built-in functions and methods that we have seen in class.
2- The program will display the main menu following the completion of each option and until the user selects option 5.

Testing Your Program:

The program does not store any data between runs. It is a good idea that you create a few sets of data that you can use to test your program and always use the same data. You can store your data in a comment at the beginning of your program such that you can copy and paste the data for input rather than typing. The more records you use to test the better.

Anne,234,20,18,12,10,15,11
Bob,124,15,18,12,15,15,12
Greg,124,15,18,12,15,13,12