

Lab Exercise #6

Assignment Overview

This lab exercise provides practice with lists and functions in Python.

You will work with a partner on this exercise during your lab session. Two people should work at one computer. Occasionally switch the person who is typing. Talk to each other about what you are doing and why so that both of you understand each step.

Programming with Lists and Tuples

Develop a Python program that will calculate and display information about exam scores for the students in a class, as described below.

The program will read the file `scores.txt` (no error checking or prompting).

Each line of the input file will represent one student and four exam scores, and will have the following format:

Name (string, maximum of 20 characters: two words separated by one comma)
Exam #x score (integer, range 0 to 100)

The student's name will be in the first 20 characters of the line. The four exam scores will be separated by one or more blanks. For example:

Hopper, Grace	100	98	87	97
Knuth, Donald	82	87	92	81
Goldberg, Adele	94	96	90	91
Kernighan, Brian	89	74	89	77
Liskov, Barbara	87	97	81	85

The program will assume that the input file contains no erroneous data.

The program will read the contents of the input file and store the data set in a **list of tuples**, where each tuple will represent one student and will contain the following information:

Name (str)
Exam #1 score (int)
Exam #2 score (int)
Exam #3 score (int)
Exam #4 score (int)
Exam average (float)

The type of each field within the tuple is listed: the student's name will be type `str`, the four exam scores will be type `int`, and the exam average will be type `float`.

After reading and storing the data set, the program will display the following.
Each line of student data will use this format string:
`"{:20s}{:6d}{:6d}{:6d}{:6d}{:10.2f}"`

A table representing the data set, **sorted alphabetically by name**

The final row will have the average of each exam with one decimal place.

Name	Exam1	Exam2	Exam3	Exam4	Mean
Goldberg, Adele	94	96	90	91	92.75
Hopper, Grace	100	98	87	97	95.50
Kernighan, Brian	89	74	89	77	82.25
Knuth, Donald	82	87	92	81	85.50
Liskov, Barbara	87	97	81	85	87.50
Exam Mean	90.4	90.4	87.8	86.2	

★ **Demonstrate your completed program to your TA. On-line students should submit the completed program (named “lab06.py”) for grading via the Mimir system.**

Suggestions:

Develop the program incrementally:

- Start by reading and displaying each line of the input file to make sure you are reading the data set correctly.
- Use the `split` string method to extract information from each line into a list. Print the list to prove that this step is working correctly.
- Convert the exam scores to type `int` and calculate the student’s average. Display those items to prove this step is working correctly.
- Create a tuple containing the six items for each student (name, exam scores, exam mean). Display the tuples to prove this step is working correctly. (Optionally, you may want to have the exam scores in a list so your tuple is (name, scores_list, mean).)
- Append each tuple to a list. Display the list to prove this step is working correctly.
- Use the `sort` list method to re-order the tuples in the list. Display the list to prove this step is working correctly.
- Use a `for` statement to display the contents of the list as a table (with appropriate formatting).
- Use a `for` statement to calculate the average of all scores on Exam #1, then display the results. Note that you could have calculated this average within the first loop, but we are **explicitly requiring** you to do this calculation by looping through your list of tuples.
- Add the logic to calculate the average of all scores on Exam #2, then display the results.