

CS160 Computer Science I

Lab 10

Objective

Work with files

Working with formatted output

Work with dictionaries

Assignment

Part 1

Write a program to ask for a series of class names and the number of credits for that class. Ask for class names until the user enters a blank value. Every class name will consist of the department followed by the class number. No other info should be present. Examples might be “CSCI 160”, “MATH 165”, or “ENGL 130”.

For every valid class name ask for the number of credits for that class. The credit hours will be integer values. Fill a dictionary with this data. Once the user has finished entering data, ask for a file name. You may use `input` or `FileUtils` to ask for the file name. Write out each part class name/credits pair in the dictionary to the data file, one pair per line. Separate the class name and credits with a tab (“\t”).

Part 2

Write each of the following functions. The function header must be implemented **exactly** as specified. Write a main function that tests each of your functions.

Specifics

In the main function ask for a filename from the user and call `readClasses`. This will return a dictionary with the information from the file. Each file should be in the formation specified in Part 1. **Do not** create this data file in this program. Then call each of the required functions and then display the results returned from the functions. Remember that the functions themselves will not display any output unless specifically stated that they should do so.

If there are built-in functions in Python that will accomplish the tasks listed below **YOU CANNOT USE THEM**. The purpose of this lab is to get practice working with dictionaries, not to get practice searching for methods.

For each function that returns a numeric value determined from the dictionary, *unless otherwise stated in the function*, if you are unable to determine a value return `None`.

Required Functions

`def readclasses (fileName)` – Creates a dictionary in the function and fills the dictionary with the class name/credits from the `fileName` file. Then it returns the dictionary. No error checking is required.

`def totalCredits (theDictionary)` – Returns the number of total credits the student is taking this semester.

`def classesByDepartment (theDictionary, department)` – Returns a list of the class names from the specified department. The list should be created in the function. Return an empty list if there are no classes from the specified department. An example of calling this function might be:
`csciClasses = classesByDepartment (classesDict, "CSCI")`

`def classesByCredits (theDictionary, credits)` – Returns a list of the class names with the specified number of credits. The list should be created in the function. Return an empty list if there are no classes meeting the specified value. An example of calling this function might be:
`fourCreditClasses = classesByCredits (classesDict, 4)`

`def isFullTime (theDictionary)` – Returns True if the student is taking 12 or more credits, otherwise this function returns False.

`def numOfClasses (theDictionary)` – Returns the number of classes the student is taking this semester.

`def upperLevelCredits (theDictionary)` – Returns the number of upper-level credits the student is taking this semester. A class is considered an upper-level class if the course number is 300 or higher.

`def printClasses (theDictionary)` – This function WILL write to the display a table of each class name and its number of credits. Include column headers in the output. Make sure the columns are neatly aligned, with the class name column being left justified and the credit column being right justified. Ensure the class names are in alphabetic order when printed. After printing all of the classes, print the total number of credits for the semester. Use the function specified above to determine this value. This value, with the appropriate label (something like "Total Credits") should have the same alignment as the rest of the information. This function should not return a value.

Questions?

Ask, sooner is better than later.