Lab 10-2 – Read CSV

Goals

- Practice with variables
- Practice with functions
- Practice file reading
- Practice file writing

Setup

- In PyCharm, create a new project or open an existing one (such as Labs)
- Create a new Python file using the following naming convention:
  `ITP115_L10_2_LastName_FirstName`
  (replace *LastName* with your last/family name and *FirstName* with your first name)
- Your new file must begin with comments in the following format (replace the name and email with your actual information):

```
# Name, USC email
# ITP 115, Spring 2020
# Lab 10-2
```

Requirements

Your program must perform the following:

- Write a program that has the following functions: **`readFile()`**, **`writeFile()`**, and **`main()`**.
- Your goal with this program is to read a CSV file and find the people under 30.
- You will be given a CSV file containing information about a person on each line.
- You will write the names of the people under 30 into a text file.

- Write the following functions:
  - **`readFile(fileName)`**
    - Parameter (1): the file name of the dictionary and set its default value to "people.csv"
    - Returns: a list containing the names of the people under 30 years old
    - Read in each line of the CSV file and split the line.
    - Check the age. If it is less than 30, then save the name to a list.
    - To split the line, use **`string.split(separator)`** with the separator set to a comma (i.e., **`","`**). It will return a list of strings.
    - Close the file object.
    - Return the name list.
  - **`writeFile(outputList, fileName)`**
    - Parameter (1): outputList is a list of names
    - Parameter (2): fileName is the name of the file to write to and set its default value to "output.txt"
    - Returns: none
    - Side effect: print a message to the user with the number of names
    - Open the file for writing.
    - Loop through the list and print each name to the file.
    - Close the file object.
  - **`main()`**
    - Ask the user for the name of the file to read from and store it in a variable. The user may enter only the return key.
    - Ask the user for the name of the file to write to and store it in a variable. The user may enter only the return key.
    - If the user entered a non-empty string for the CSV file, then call the **`readFile`** function with the file name. If not, then call the **`readFile`** function without a file name.
    - Calling the **`readFile`** function will return a list of names, so make sure to save that into a variable.
    - Call the **`writeFile`** function (with or without the name of the output file depending on the user's input) to save the list to the output file.

- Call the **`main`** function.

Sample Output 1

```
Enter the name of the CSV file you wish to read in: people.csv
Enter the name of the file you wish to write to: results.txt
3 names were written to results.txt
```

Sample Output 2 (using default values by user pressing enter/return key)

```
Enter the name of the CSV file you wish to read in:
Enter the name of the file you wish to write to:
3 names were written to output.txt
```

Deliverables and Submission Instructions

- Create a zip file containing your Python code. This cannot be done within PyCharm. Find the file or folder on your computer and compress it.
  a. Windows*:*
     1. Using File Explorer, select your lab file
     2. Right click
     3. Send to ->
     4. Compressed (zipped) folder
  b. Mac OSX:
     1. Using Finder, select your lab file
     2. Right click
     3. Compress "*FileName*"
- Upload the zip file to your Blackboard section:
  1. On Blackboard, click on the Labs item in the course menu on the left.
  2. Click on the specific item for this assignment (starts with L and a number).
  3. Click on the Browse My Computer button and select your zip file.
  4. Click the Submit button.