*Turn on the Navigation Pane in Word when viewing this document:* **View tab > enable Navigation pane**

# Program Specification

* *Program title:* *Customer File Menu*
* *File name:* MIS3301-Ch6-HW-CustomerFileMenu
* *Due Date:* see Canvas for due date
* *Main objective:* *Victorino Insurance Company* is an insurance company . They are requesting that you write a program that allows their insurance agents to enter basic customer data in a CSV file. The application must be menu-driven and allow the user to: 1) view a list of all customers, 2) search for a customer by ID, 3) enter a new customer, 4) update a customer, and 5) delete a customer.

# Read before starting

* The expectation is that students have read the *entire* chapter and attended *all* lectures before beginning this assignment.
* Reminder: This is an *individual* assignment. Students are expected to complete this assignment independently and without the assistance of others. If you need help, contact your instructor via email or visit during office hours.
  + When emailing your instructor, please provide as much information as possible:
    1. State what the issue is and provide line numbers.
    2. Attach your \*.py file *zipped* as outlook does not permit attaching these files.
    3. Also, copy & paste the snippet of code where the issue is occurring within the email. If I am not at a computer when I see your email, I may be able to respond by seeing the snippet of code.
* I recommend that you read the entire assignment before beginning.
* Periodically review the ***Assignment Tips & Updates*** page in our module looking for tips as well as any ***required*** updates to this assignment.
* **All of your code for this assignment must be written in functions.**
* **Globals** - global constants are permitted but global variables are not.
* **This code must be free of syntax errors, or it will not be graded. Thus, comment out any code that crashes, or for a better grade, improve the code and submit it one day late with a 10% deduction.**

# Instructions

## Getting Started

* Open IDLE > open the *Editor Window* (File >NEW)
* Click File > Save As…
* Locate your Chapter 6 folder for this course
* Save your file as: **MIS3301-Ch6-HW-CustomerFileMenu.py**

## Add Comment Lines

* Copy the comments lines below to your code.

|  |
| --- |
| #add required assignment header comment lines here  #Import Modules  #Global Constants  #MAIN--------------------------------------------------------------------------------------  #FUNCTIONS---------------------------------------------------------------------------------  #Call main() here |

## Download the files

* Open your Canvas assignment and download the following files **to the same folder as your assignment:** customers.csv, customers-BKUP.csv - this file should never be updated as it serves as your original backup data. Do not confuse this with TEMPFILE referenced in some of the lecture slides.

## At the top of your code:

* Add code to import the **csv** and **os** modules.
* Add 3 global constants at the top of your code for the customer file, backup file, and temp file: **CUSTFILE, BKUPFILE,** and **TEMPFILE**. Thus, file names should not appear in the rest of your code as you will be using these constants. Set these constants to the file names that you downloaded. However; for the TEMPFILE, name it as: customers-TEMP.csv. Recall, the TEMPFILE is used when updating files and will be created dynamically in your code and then deleted.

## Code all Functions Headers

* Add your **main()** function header. Add only a pass statement for now – but remove it later.
* Add all **other** **function headers** (Fn1-5, Fn7).
  + Name the functions appropriately based on the menu choices shown in the Main Menu screenshot below. Make sure to start each function name with a verb!
  + Add comment lines such as #Fn1, #Fn2, …#Fn7 above each function. There is no #Fn6.
  + **Begin starter code for each function**. For now, type a single print statement in each function so that you can test that the menu is working in the next step - e.g. in function 1, type print(“….Option 1 was executed”).
  + Leave 1 to 2 blanks lines between functions.
  + FYI: None of the functions in this assignment will have a return value; these functions are mainly for organization of your code and not reusability as such you will be opening & closing files and writing print statements in each.
* I recommend that you code Fn7 now as it contains only 2 lines of code, and the code is already provided for you – see step 7 below.
* Call the main() function at the bottom of your code.

## Main Menu

* Inside of your main() function, display the menu below. Allow the user to enter a choice and based on their choice invoke the correct function. Place this menu code in a *while* loop to allow the user to continue making choices. The loop should run as long as the user has not entered 99 to exit. This will allow the user to press 1 to view the customer list, or use any other option until they are ready to exit the application. Do not change the numbers in the menu (as these will be expected when grading).

Graphical user interface, text, email

Description automatically generated

* **Ensure that the main() function simply handles displaying the menu, prompting the user for a menu choice, and calling the appropriate function.** All other logic will be coded within functions. Thus, you will not be opening the customer file within main() as eachunction will handle opening the file in whatever manner it requires (i.e., for reading or writing) as well as closing the file when it is done.
* Handle basic validation: display an error message if an invalid choice is entered and allow the user to enter another choice.
* **Test the menu fully before proceedin**g. For example:
  + Execute your code and press 1.
  + Make sure that it invokes the function which should display “….Option 1 was executed”. It should then loop back to the top and re-displays the menu. If it does not display the menu again, your loop is not working.
  + Make any corrections and continue testing all options.
  + Once this is working, begin coding the rest of the functions.

## **Note: you will be opening & closing files in each function. Do not open the files in main().**

## Fn1 - View customer list

* Write a function to display all customers in the file as shown below. When opening the CSV file, make sure to use the constant not the actual file name. As usual, make sure this code displays all data as shown in the screenshot below including the correct number of total records. Keep in mind that your program will be graded using another set of test data. Achieve this step in its entirety before moving to the next step.
* **Note:** This function will execute when the user selects menu option 1 to display the list of customers. The first line in the output screenshot below is the last line of the main menu screenshot shown above; do not prompt the user to enter a menu choice in this function as that should only occur in main() – this note is also true for all of the remaining functions.

Table

Description automatically generated

## Fn2 - Lookup a customer

* Write a function that allows the user to enter a customer ID, then retrieve the information for that customer from the customer file. Perfectly align the field values as shown below. Also, let them know if a record was not found. Achieve this step in its entirety before moving to the next step.

Text

Description automatically generated

## Fn3 - Enter new customer

* Write a function that allows the user to enter a new customer and add this data to the bottom of the customer file. Perfectly align the data entry values as shown below. Achieve this step in its entirety before moving to the next step. You can test if the data was properly added by going back to menu option #1 to re-display the customer list (or by opening the CSV file).
* Handle basic data validation: if all 3 values are not entered, notify the user and do not save the record.
* **Next, I recommend that you write Fn7-Copy customer data from BKUP below**. In the event that your data is destroyed, you can easily copy it back with option 7. If you somehow delete both the real file & the backup file, just grab a new copy from Canvas.

A picture containing graphical user interface

Description automatically generated

## Fn4 - Update a customer

* Write a function that allows a user to update an existing customer’s data in the customer file. Require the user to update at least one field. In this example, the user only updated the last name. Notify the user if the record was updated. Also, notify them when a record was not found. Tip: if they leave a value empty, make sure to write out the original value. Achieve this step in its entirety before moving to the next step. You can test if the data was properly updated by going back to menu option #1 to re-display the customer list (or by opening the CSV file).

Text

Description automatically generated

## Fn5 - Delete a customer

* Write a function that allows a user to delete an existing customer from the customer file. *Upon entering a customer ID*, display the customer data that was found and prompt the user for confirmation that they want to delete this customer. Notify the user if the record was deleted. Also notify them when a record was not found. Achieve this step in its entirety before moving to the next step. You can test if the data was properly deleted by going back to menu option #1 to re-display the customer list (or by opening the CSV file).

Graphical user interface, text, application, email

Description automatically generated

## Fn7 - Copy customer data from BKUP

* 10 points will be deducted if this option is not included and working properly.
* **You will only 3 lines of code for this function.** The following two lines plus a message telling the user that the data was recovered from the backup file.

from shutil import copyfile

copyfile(BKUPFILE, CUSTFILE)

* This code is useful for testing purposes. It copies the backup file to the customer file in order to restore it. Since you will be entering, updating, and deleting records while you are testing your code, you will want to reset the customers.csv file back to the original data. You can run this as many times as you want.
* NOTE: the BKUP file should never be updated as it serves as your original backup data.
* Recall that BKUPFILE and CUSTFILE are global constants defined at the top of your code.

# Submitting the Assignment

* First, review the ***Assignment Tips & Updates*** page to see if there are any additional requirements to this assignment – denoted with the tag: **[required].**
* To submit the assignment, go to Canvas and open the assignment link.
* Upload your \*.py file**.** You may upload more than once as long as it is before the due date & time. Only the final submission will be graded. Assignments will be accepted one day late albeit with a 10% deduction.
* **Note:** You will only submit your program since your code will allow us to create a new file and add records. Do not upload any csv files.