(CH7) Program Assignment Instructions

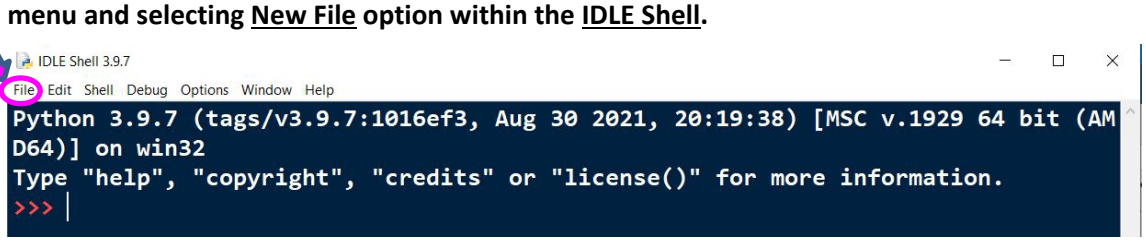
Last Changed: 4/15/2023 5:26 PM

Read and follow the directions below carefully and perform the steps in the order listed. You will be solving one program as instructed and turning in your work electronically via an uploaded file within Eagle Online/Canvas. Make sure and check your work prior to uploading the file.

*Note: Refer to **(SET) How to Download Install and Use Python IDLE (Windows User)** file (**Page 8**) and/or **Use Python IDLE Video** link within **Module 2** on how to **create**, **enter**, **save**, **run**, and **submit** a **script (source code/program) file**.*

Instructions

1. **Using Python IDLE, create a New Empty Script (Source Code/Program) File in your working drive by clicking the File menu and selecting New File option within the IDLE Shell.**



2. **Save the new Script (Source Code/Program) File with the name CH7LastFirst.py by choosing File menu's Save option *(Note: Make sure to know where you saved the file in your working directory for future submission)***

   *(NOTE: Where **LastFirst** should be replaced with your actual **Lastname** and **Firstname**. For Example, if your name is Mary Smith then your file name should be named: **CH7SmithMary.py.** Reminder: You don't need to add the **.py** extension, IDLE will automatically add the extension)*

3. ***You will develop a modular Number Analysis Program (using functions, lists, loops, and selections) as described below:***

   Write a **modular** program *(using **functions**)*, that accepts from the user a series of 5 numbers *(using a **loop**)* and stores the numbers in a **list**. The program should then determine the **lowest** number, **highest** number, **sum** of the numbers, and **average** of the numbers in the list calling appropriate functions as described in **step 4b *(do not use readily available standard functions like min, max, sum, sort, etc.!)***. Refer to the table in step 5 below for various sample runs and the corresponding output displayed.

4. a) The first thing you should enter in your python program script file is a top comment block which includes the following:

```
# Name: Enter your full name here
# Date: Enter today's date here
# Program: Chapter 7 – Lists
# Description: Enter a paragraph description of the program
```
(at least 5 sentences in your own words and do not copy from the program description above)

b) Below the top comment/documentation block, type the actual Python code for the problem described in step 3 ***(Refer to the below descriptions of the functions to include and Source Code screenshot with example statements and HINTS):***

i) **`main()`**

This should be the first and foremost function called and defined *(Refer to line #s 70 and 10 - 17 in the below sample code screenshot on how this function is called, and HINTS for how other functions should be called in the main function definition)*

ii) **`getNums()`**

This should be a **value returning** function **called by main function**, that accepts **no arguments**. This function asks the user for 5 numbers and stores in a list and returns the list back to the main function *(HINT: Refer to line #s 14 and 20 - 27 in the below sample code screenshot)*

iii) **`findLo()`**

This should be a **value returning** function **called by main function**, that **accepts one list argument**. This function should find and **return the lowest** of the numbers in the list back to the main function. Make sure to not use min or sort functions *(HINT: Refer to line #s 14 and 20 - 27 in the below sample code screenshot)*

iv) **`findHi()`**

This should be a **value returning** function **called by main function**, that **accepts one list argument**. This function should find and **return the highest** of the numbers in the list back to the main function. Make sure to not use min or sort functions *(HINT: Refer to line #s 14 and 20 - 27 in the below sample code screenshot)*

v) **`findTotal()`**

This should be a **value returning** function **called by main function**, that **accepts one list argument**. This function should find and **return the total** of the numbers in the list back to the main function. Make sure to not use min or sort functions *(HINT: Refer to line #s 14 and 20 - 27 in the below sample code screenshot)*

vi) **`findAvg()`**

This should be a **value returning** function **called by the main function** that accepts **one total argument**. This function should find and **return the average** back to the main function *(HINT: Refer to line #s 14 and 20 - 27 in the below sample code screenshot)*

vii) **`displayData()`**

This should be **a void** function **called by main function** that accepts **four arguments *(lo, hi, total, and avg)*** and display them *(HINT: Refer to line #s 14 and 20 - 27 in the below sample code screenshot. Make sure **avg** variable is formatted up to 2 decimal places and displayed)*

c) ***Make sure to include the following in your code:***

i) Use **CONSTANTS** appropriately. Generally, we use all caps (capital letters) for **CONSTANTS** with underscores separating the words. (for example: LIST_SIZE)

ii) Use descriptive and appropriate **identifiers** (variable and function names) with **smallCamelCase** naming style. (for example: numsList, getNums(), findLo(), findHi(), etc.)

iii) Keep proper **documentation** for **understandability of your program** by adding **block comments** with paragraph description prior to each function explaining the function logic *(refer to EXAMPLE PROGRAMS listed in Canvas modules and in Revel readings)*.

iv) Use proper **indentation** and **blank/line spaces** in your program for **readability of your code/program** *(refer to EXAMPLE PROGRAMS listed in Canvas modules and in Revel readings)*.

Refer to program screenshots below for HINTS:

CH7RaoSuma.py - C:\COSC1436\CH7RaoSuma.py (3.9.5)

File Edit Format Run Options Window Help

```
# Name: Enter your full name here
# Lab: Chapters 7 - Lists
# Description: Enter a paragraph description of the program (at least 5 sentences)

# CONSTANTS
TITLE = "Welcome to Number Analysis Program!\n"
LINE = '-' * len(TITLE)
LIST_SIZE = 5

# main function description goes here...
def main():
    print(TITLE+LINE)
    numsList = []
    numsList = getNums()
    lo = findLo(numsList)
    # Complete remaining function calls below:



# getNums function logic/description goes here...
def getNums():
    numsList = []
    # Complete remaining statements following the HINTS below:
    # HINT: Use a for loop ranging 0-LIST_SIZE
        # HINT: Input an integer number
        # HINT: Append the number input to numsList
    # HINT: Make sure to return back numsList

# findLo function logic/description goes here...
def findLo(numsList):
    lo = numsList[0]
    # Complete remaining statements following the HINTS below:
    # HINT: Use a for loop iterating through every element in numsList
        # HINT: Use if to check if the element is less than lo variable
            # HINT: Then, assign the list element to lo variable
    # HINT: Make sure to return the lo variable

# findHi function logic/description goes here...
# Follow above function HINTS and complete findHi function definition below:




# findTotal function logic/description goes here...
# HINT: Define findTotal function header
    # Complete remaining statements following the HINTS below:
    # HINT: Assign 0 to an accumulator variable total
    # HINT: Use a for loop iterating through every element in numsList
        # HINT: Accumulate to total each element in the numsList
    return total

# findAvg function logic/description goes here...
def findAvg(total):
        # HINT: Make sure to return the expression of calculating average

# findTotal function logic/description goes here...
# HINT: Define displayData function header
    print(LINE)
    print("Lowest Number =", lo)
    # HINT: Print hi, total, and avg (formatted to 2 decimals) below:



    print(LINE)

# main function called below:
main()
```

5 After completing your program as instructed, make sure to **Run** your program/script file to obtain the output/results as shown in the sample run screenshots in the table below for various inputs:

1.

```
>>>
====================== RESTART: C:/COSC1436/CH7RaoSuma.py
Welcome to Number Analysis Program!
-----------------------------------
Enter a number: 5
Enter a number: 6
Enter a number: 11
Enter a number: 2
Enter a number: 3
-----------------------------------
Lowest Number = 2
Highest Number = 11
Total of Numbers = 27
Average of Numbers = 5.40
-----------------------------------
>>>
```

2.

```
>>>
====================== RESTART: C:/COSC1436/CH7RaoSuma.py
Welcome to Number Analysis Program!
-----------------------------------
Enter a number: 1
Enter a number: 1
Enter a number: 3
Enter a number: 3
Enter a number: 2
-----------------------------------
Lowest Number = 1
Highest Number = 3
Total of Numbers = 10
Average of Numbers = 2.00
-----------------------------------
>>>
```

3.

```
>>>
====================== RESTART: C:/COSC1436/CH7RaoSuma.py
Welcome to Number Analysis Program!
-----------------------------------
Enter a number: 5
Enter a number: 5
Enter a number: 5
Enter a number: 5
Enter a number: 5
-----------------------------------
Lowest Number = 5
Highest Number = 5
Total of Numbers = 25
Average of Numbers = 5.00
-----------------------------------
>>>
```

6. You may now proceed to Program Assignment INSTRUCTIONS and UPLOAD link within this module and follow the steps in the link or follow the steps below to submit your work as a **File Upload** (an attached **.py** file):

    a. Choose the **Start Assignment** button,
    b. Choose **File Upload** tab,
    c. Choose **Browse** to locate your **script (source/program) file** to add,
    d. Choose **Submit Assignment** to complete file upload.

NOTE: ONE OF THE COMMON MISTAKES IS THAT STUDENTS ENTER PYTHON COMMANDS/STATEMENTS IN THE "IDLE SHELL" DIRECTLY AND SAVE THE RESULTS TO A FILE AND SUBMIT WHICH IS INCORRECT!!!

INSTEAD...

YOU SHOULD FOLLOW THE ABOVE STEPS TO CREATE A NEW SCRIPT (SOURCE CODE/PROGRAM) FILE FROM THE IDLE SHELL, SAVE THE FILE, ENTER PYTHON STATEMENTS (PROGRAM) INTO THE FILE, RUN YOUR PROGRAM, AND SUMBIT THAT SCRIPT (SOURCE CODE/PROGRAM) FILE AND NOT THE OUTPUT OF THE IDLE SHELL!!!