Homework 04

COMSC-122

Fall 2017

Homework-4A: Input Validation Using a While Loop

- In Chapter 03, we studied the Grade Program, which assigned a letter grade based on the numeric scores that students received in a quiz.
- What wasn't done was to check whether the scores typed in were greater than 100 (an obvious inputting error).
- Nor did we test to see if someone accidently typed in a negative number (another obvious inputting error).
- We also had to re-run the program each time we wanted to convert another quiz score to a letter grade.
- Now we'll correct these three deficiencies.
- Shown below is the Elif version of the Grade Program:

Elif Version of Program 3-6

```
# Grader2.py
# This program gets a numeric score from the user and displays the corresponding letter grade
# Your Name
A_score = 90
B_score = 80
C_score = 70
D_score = 60
print(“Your Name’s Grading Program”)
score = int(input('Enter your test score:'))
if score >= A_score:
        print('Your grade is A.')
elif score >= B_score:
        print('Your grade is B.')
elif score >= C_score:
        print('Your grade is C.')
elif score >= D_score:
        print('Your grade is D.')
else:
        print('Your grade is F.')
exit = input('')
```

Homework-4A: Input Validation Using a While Loop

- Your task is to embed this program into two while loops.
  - One While loop will ask if you would like to convert another quiz score to a letter grade, which will give you the option of running the program again without having to restart it, or not.
  - The second While loop will test the score that you type in to be sure that it is >= 0 and <= 100.
    - If it is not in this range, then your loop will ask you to re-enter your data.
    - Otherwise the program may proceed.
  - Be sure your name appears on the screen in the output
  - Name your program: *YourName*-Hwrk04A.py and submit it to the dropbox.

Homework-04B: Uses of for Loop

- We would like an output that looks exactly like that below with your name at the top as shown:

Sales Bar Chart

Write a program that asks the user to enter today's sales for five stores. The program should then display a bar graph comparing each store's sales. Create each bar in the bar graph by displaying a row of asterisks. Each asterisk should represent $100 of sales.

Here is an example of the program's output.

***Your Name*'s Store Sales**

```
Enter today's sales for store 1: 1000 [Enter]
Enter today's sales for store 2: 1200 [Enter]
Enter today's sales for store 3: 1800 [Enter]
Enter today's sales for store 4: 800 [Enter]
Enter today's sales for store 5: 1900 [Enter]

SALES BAR CHART
(Each * = $100)
Store 1: **********
Store 2: ************
Store 3: ******************
Store 4: ********
Store 5: ********************
```

Homework-04B: Uses of for Loop

- You may assume that all sales data is rounded to the closest dollar (i.e. all sales data may be considered to be integers).
- Tip: If you don't want your print() statement to start a new line after its execution, then be sure that the last part of the print statement contains:
  - **, end = ")**
  - For example: **print('*', end=")** will not cause the next print statement to start on a new line.
  - **print('*')** will cause the next print statement to start on a new line.
  - You will need both of these constructs to do this exercise.
- There is an opportunity to use both types of for loops that we have been introduced to in this problem
- Please name your solution: *YourName*-Hwrk04B.py