Assignment 5 – For Loops

Learning Objective

Write a Python program that uses `for` loops.

Assignment Description

Write a program that allows the user to enter a sentence and then uses looping to print an asterisk (*) to show the character distribution.

Steps

1. In PyCharm (Community Edition), open your existing ITP115 project.
2. Under the Assignments directory, create a new directory called **a5_*last_first*** where *last* is your last/family name and *first* is your preferred first name. Use all lowercase letters.
3. In the directory, create a new Python file called **assignment5.py**.
4. At the top of the file, put comments in the following format and replace the name, email, and section with your actual information:

```
# Name, USC email
# ITP 115, Fall 2022
# Section: number or nickname
# Assignment 5
# Description:
# Describe what this program does.
```

5. Create a variable (str) that contains the letters of the alphabet.
6. Get input from the user.

```
Enter a sentence: You miss 100 percent of the shots you never take.
```

7. You want to count the number of special characters, which are characters that are not in the alphabet and not spaces. Numbers will be special characters. Create a variable and use a `for` loop to loop through the user's input. You will also need to use branching.

8. Print out the number of special characters using asterisks. You can use the multiplication operator with a string and number to repeat the string. If there were no special characters, then print the word NONE.

```
Special characters: ****
```

9. Print out the number of character distribution for the letters of the alphabet. Loop through the variable containing the letters of the alphabet. Create a variable for counting the number of times each letter occurs. Use a nested loop to loop through the user's input and use branching to check if the character matches the alphabet letter. Make sure that you handle having the user input upper case and lower case letter. If "A" and "a" were in the user's input, then the count for "a" is 2. If a letter did not appear in the user's input, then print the word NONE.
   Important: you must use a loop to count; you may **not** use the **`count()`** function.

```
a: *
b: NONE
c: *
d: NONE
e: ******
```

10. Update the program to allow the user to enter in how many times to run the program at the beginning. Use a range-based for loop to accomplish this.

```
Character Distribution
Enter the number of times to run: 2
```

11. Be sure to comment your code. This means that there should be comments throughout your code. Generally, a comment for every section of code explaining what it does. Points will be deducted for not having comments.
12. Follow coding conventions. You should use lowerCamelCase or snake_case for variable names. You are welcome to create any variables that you need.
13. Test the program. Look at the Sample Output below. When testing, the graders will only enter integers for the number of times to run the program. Assignments that do not run are subject to 20% penalty.
14. Prepare your submission:
    - Find the **a5_*last_first*** folder on your computer and compress it. This cannot be done within PyCharm.

- On Windows, use **File Explorer** to select the folder. Right click and select the Send to -> Compressed (zipped) folder option. This will create a zip file.
- On Mac OS, use **Finder** to select the folder. Right click and select the Compress "*FolderName*" option. This will create a zip file.

15. Upload the zip file to your Blackboard section:
    - On Blackboard, navigate to the appropriate item.
    - Click on the specific item for this assignment.
    - Click on the **Browse Local Files** button and select the file.
    - Click the **Submit** button.

Grading

- This assignment is worth 30 points.
- Make sure that you the program runs. Points will be taken off if the graders have to edit the source code to test your program.
- Make sure to submit your assignment correctly as described above. Points will be taken off for improper submission.

Item	Points
Run program using a range-based for loop	5
User input for sentence	2
Count number of special characters	5
Print asterisks or NONE for special characters	4
Loop through alphabet using str variable	5
Count number of each letter	5
Print asterisks or NONE for each letter	4
Total	**30**

Sample Output

```
Enter the number of times to run: 2

Enter a sentence: You miss 100 percent of the shots you never take.
Special characters: ****
a: *
b: NONE
c: *
d: NONE
e: ******
f: *
g: NONE
h: **
i: *
j: NONE
k: *
l: NONE
m: *
n: **
o: ****
p: *
q: NONE
r: **
s: ****
t: ****
u: **
v: *
w: NONE
x: NONE
y: **
z: NONE

Enter a sentence: Python is fun
Special characters: NONE
a: NONE
b: NONE
c: NONE
d: NONE
e: NONE
f: *
g: NONE
h: *
i: *
j: NONE
```

```
k: NONE
l: NONE
m: NONE
n: **
o: *
p: *
q: NONE
r: NONE
s: *
t: *
u: *
v: NONE
w: NONE
x: NONE
y: *
z: NONE
```