Assignment 2 – Mad Libs Story

Learning Objective

Write a Python program that creates and uses variables of different types. Get and store input from the user. Output information including escape characters.

Assignment Description

Create your own Mad Libs story. The idea of the Mad Libs game is to write down different words that fit parts of speech, and then incorporate those words into a prewritten story.

Steps

1. In PyCharm (Community Edition), open an existing project (such as ITP115) or create a new project.
   - If you open an existing project, then create a new directory (probably under the Assignments directory) named **a2_*last*_*first*** where *last* is your last/family name and *first* is your preferred first name. Use all lowercase letters.
   - If you create a new project, then name it **a2_*last*_*first*** where *last* is your last/family name and *first* is your preferred first name. Use all lowercase letters
2. In the project or directory, create a new Python file called **assignment2.py**. 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, Spring 2022
# Section: number or nickname
# Assignment 2
# Description:
# Describe what this program does such as:
# This program creates a Mad Libs story.
# It gets input from the user and prints output.
```

3. Get user input for the following input and store them in variables:
   - 5 strings (**`str`**)

- 3 integers (**`int`**)
- 1 floating point number (**`float`**)
- Name the variables whatever you want, but use good coding practice.
- You can get more user input if you want.
- The strings need to be different parts of speech such as a noun, verb, adjective, adverb, or preposition. You can also ask for a special type of the part of speech, such as a place ("Disneyland"), a proper name ("Khurram"), or a color.

4. At least two of the values entered must be in used in some mathematical calculation (e.g. adding, multiplying, etc.), and the result should be used in the story. Make sure to use the int() and float() functions to convert the strings entered by the user to the appropriate number type.
5. Your program must then print out the story with the user's words injected into the story. You may use multiple print() statements. If you do not use multiple print() statements, then use the new line escape character to print on multiple lines.
6. You need to signify to the user what their words are in the story using "" (e.g. "chicken" or "42"). To do this, use escape characters.
7. Since there will quotation marks around the words, follow proper formatting and ensure there are no extra spaces (e.g. "42", not " 42 ").
8. Create your own story. Do not use the story in the Sample Output.
9. 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.
10. Follow coding conventions.
11. Test the program. Look at the Sample Output below. Assignments that do not run are subject to 20% penalty.
12. Prepare your submission:
    - Find the **a2_*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.

13. 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 zip file.
    - Click the **Submit** button.

Grading

- This assignment is worth 25 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
Reading in each individual input (9)	9
Mathematical calculation	2
User input is in the story with surrounding quotes	9
Comments, style, and proper submission	5
Total	**25**

Sample Output

```
Enter an animal (plural): sloths
Enter an adjective: funny
Enter another adjective: hungry
Enter a verb: fly
Enter a verb ending in 'ing': playing chess
Enter a number: 3
Enter a second number: 10
Enter a third number: 4
Enter a number with a decimal: 3.2

Today I adopted "3" pet "sloths".
I learned that each animal needs "3.2" hours of "playing chess"
every day, and that they travel in groups of "4".
They are so "funny" that I decided to adopt "10" more.
Now I have "13" "sloths" and I am so "hungry" that I want to "fly".
```