Exercise 2: First small Python Programs

Programming Techniques in Computational Linguistics 1
Programming for Linguists

This is a non-graded exercise. We recommend that you solve it and hand it in by 25. October 2023 at 6 pm. If you submit it in time, we will give you feedback.

Remarks on submission:

- Please submit the exercises in groups of two. If you can't find a partner, use the OLAT forum or write us an e-mail.
- Accepted submission file formats: `.txt/.py/.pdf/.zip`
- File naming convention in general: `olatusername1_olatusername2_pcl1_exercise0X.txt/pdf/py/zip`.
  - Example: `jbiden_dtrump_pcl1_exercise03.zip`
- Write many comments! They don't just help the tutors to understand what you're thinking, but can also help you to find and fix possible bugs.
- Please number the tasks on the submission sheet/Python programs the same as on the task sheet.

1 Python in the Terminal

In your terminal emulator (**WSL** on Windows, **Terminal** on macOS), open the Python interpreter with the command `python3`. Try out some of the functions shown in the lecture and do the following:

- Define a variable and modify its value.
- Use an `if-else` statement
- Compare two values using ==, > or <.
- Use `exit()` to leave the Python interpreter.

Note: If the `python3` command is not available, try using `python` instead.

2 Text-number Conversion

You are given 2 formulae:

$$x = 12a^2b - 9b^3a \tag{1}$$

$$y = \frac{x * (b + a)}{0.4b + 0.3a^2} \tag{2}$$

1. Write a program that prompts the user to input values a and b, then outputs the values of x and y on separate lines. The program should follow this pattern:

```
Enter the value of a: 2
Enter the value of b: 3
Value of x: -342
Value of y: -712.5
```

   The results should not be rounded.

   Hint: Python always reads user input as a string. The function `int(argument)` turns the argument into an integer if it recognizes the argument as a number, even if the argument's data type is a string (e.g. `int('8')` and `int(str(8))` would both evaluate to `8`). The `float(argument)` function turns the argument into a floating point number when it recognizes the argument as a number.

2. How does the difference between `int()` and `float()` affect the result?

Provide your solution to Part 1 in the provided file *Task_2.py*. Write your response to Part 2 as a comment at the bottom of the same file.

3 String Input and Length Comparison

Write a program that prompts the user for their favorite food and their favorite country. Then, use that information to output (print) a string in the format:
"It seems your [favorite food] is popular in [favorite country]."

- If the name of the food is longer than 7 characters (i.e. ≥ 8 characters), then it should be printed entirely in uppercase.
- If the name of the country has exactly 5 characters, print "beautiful" before the country name.

The program flow should follow this pattern:
Example 1:

```
Enter your Favorite Food:
> Souvlaki
Enter your Favorite Country:
> Greece
It seems your SOUVLAKI is popular in Greece.
```

Example 2:

```
Enter your favorite food:
> Pizza
Enter your favorite country:
> Italy
It seems your Pizza is popular in beautiful Italy.
```

Enter your solution in the provided file `Task_3.py`.

4 `if` Statement

Write a program that prompts the user for a German noun. The program should use the inputted noun's suffix to determine its grammatical gender and output its corresponding (definite) article. Your program should use the following rules to identify the gender of the user-inputted word:

- If the word ends in *-smus*, *-ner*, *-ich* or *-ist*, the article is *der*.
- If the word ends in *-heit*, *-keit*, *-schaft* or *-ung*, the article is *die*.
- If the word ends in *-chen*, *-tum*, *-lein* or *-ium*, the article is *das*.
- If none of these apply, the gender cannot be determined.

Enter your solution in the provided file `Task_4.py`.

5 `while` Loop

5.1 Task 5a: Counting Sheep

Your Computer is having trouble falling asleep, but worry not, because you can help.
Create a `while`-loop to make your computer count sheep until it falls asleep. At each iteration (execution) of the loop, add to the count and display the counting progress. The loop should be interrupted if the necessary number of sheep, `n`, has been reached and a final message containing the number of sheep the computer counted should be displayed at the end. You can wish your computer "good night"for example :)

Example output:

```
The computer is counting sheep to fall asleep!
1 sheep
2 sheep
3 sheep
4 sheep
z.. z.. z..
Your computer fell asleep after counting 4 sheep. Sweet dreams little machine!
```

Enter your solution in the provided file `Task_5a.py`. Note: The number of sheep `n` is randomly generated in the code skeleton in the provided file `Task_5.py`

5.2 Task 5b: Code Comparison and Understanding

After you've completed Task 5a, use ChatGPT to generate a solution for the same task in respect to your current programming skills and paste it in the provided file `Task_5b.py`.

1. Once you have both solutions, write down the similarities and differences between the two solutions.
2. Explain why both codes (if they are different) achieve the same result or why they might give different outputs.

It's crucial to not only get a solution but to understand the various ways a problem can be approached and solved in programming. Enter your observations, comparisons, and explanations as comments in `Task_5b.py`.

6 Comments

Comments help us understand a program. This is not only important for us tutors, but also helps you understand your own code now and even more so in the future. To illustrate how useful good comments are and to give you an opportunity to practice, we would like you to add comments to the Python script `comment.py`.

1. At the top of the file, add a description of the script's purpose/objective.
2. Add useful comments (not necessarily in every line) describing the steps of the code and the meaning of the variables. Please make sure to modify every line that already contains a comment (lines beginning with #).

To find out what a particular line in the script does, it is helpful to comment out that line and see how the program changes. Please submit `comment.py` containing your additional comments.