Advanced Technical Exercise 🌶🌶🌶

Shared Exercises 2

These are technical exercises, that guide in going through the content of the second workshop. As before, try to solve each exercise without looking at the hints, but feel free to use Google or StackOverflow or anything else as an information source. If you feel a bit stuck look at the hint. If you are still stuck, do not hesitate to reach out to me! After finishing an exercise, you might still want to read the hints - sometimes they might give you an inspiration how to approach the exercise differently.

Note, that each of these exercises (as pretty much any programming problem) has multiple correct solutions. It can be very instructive to implement several of them and discuss what the advantages and disadvantages they might have.

Exercise 1

Download the file ”python.txt” from the Exercises folder. This is a text file containing some text copied from the official python.org website. Upload this text file into Google Colab (or just put it in the same folder as your notebook). Read this file into memory, as a list of lines.

Exercise 2

Let’s process some of this text to make it easier to work with. Every one of the following bullet points can be achieved using list comprehension. Try to do it that way. Do the following:

- Look up the .lstrip() and .rstrip() methods. Use these to remove all instances of the newline character in your list of the lines of text.
- Convert all the lines into lowercase.
- Look up the ”string” library in the Python standard library. This contains many useful string constants, including a string that contains punctuation symbols. Look up the str.maketrans() method. Can you use these things together to remove all punctuation from the lines of text in the list?

Exercise 3

Take a look at the text file. Some of the lines are obviously headings of sections. Based on the lengths of these heading lines, can you extract them into a new list? Call this list headings.

Exercise 4

Create a new file called ”headings.txt” and write out the headings you just extracted into this new file.

Exercise 5

From the original list of lines of text, create a new list that contains all the individual words in this file.

Exercise 6

Make a new list called unique_words, in which you only have unique words from the list of words, i.e. no duplicates.

Exercise 7

Create a dictionary, where the keys are the unique words, and the values are the number of times they appear in the ORIGINAL word list (before you removed duplicates). Use dictionary comprehension to do this step in a single line of code.

Exercise 8

You will notice that some words appear a lot. Words like "the" or "and" and so on. These words don't contribute much meaning to text data. Also, very short words, like "in", "of" or "or", also typically do not contribute much meaning (this is of course not always the case, but let's assume so for now. So let's make a new list called meaningful_words, which only contains words that occur 3 times or less in the text, and which are at least 4 letters long.