**HW 4D**

**Learning Experience**

Part 1:

The first part of the assignment was a tricky program. The program was about to read a file .txt and then a store each word in a list. To do so, I just needed to open the file, read the file, and store the whole text in a string.

Graphical user interface

Description automatically generated

Once I have the whole text in a string, I needed to break each word up and store in a list; however, the text was compound not only with white space but also with dashes between words, so using the simple split() built-in function won’t help. Therefore, what I needed was to split the words with multiple delimiters and to do that I had to import “re” from python library which allows me to split the text with multiple delimiters.



After the previous code, some words from the list had special characters. To enhance this part of the assignment, I wanted to get rid of these special characters from the words, so I use a single line for-loop to loop from the list and to clean special characters from the words by using .strip() built-in function



However, there were still some words having contractions like he’ll or we’ll, so I also wanted to delete these in order to have the single word itself. To do so, I used the built-in function .replace() in order to replace ’ll from the words:

Text

Description automatically generated with medium confidence

The next step was to get rid of the duplicate words. So, I decided to do a for-loop instead of set for this piece of code since the last assignment I used sets. The for-loop was a tricky one because I had to find a way to compare each element of the list and determine if that element repeats in the new no\_duplicate\_list. To do so, I used the membership operator “in” to check if the word is present in the no\_duplicate\_list. So, if the word is not in the no\_duplicate\_list, append the word to the no\_duplicate\_list, otherwise do not append.

Text

Description automatically generated

The next step was to delete the words: “the, a, an” from the list. This was easy just by doing a for-loop to loop through the list and an if-statement to determine if each word is equal to ‘the’ or ‘a’ or ‘an’. If that is the case, so remove it by using the built-in function .append(word).

Finally, the last part of the code is to sort the list in ascending order. This was done by using the function .sort().

Part 2:

The second part of the assignment was the easier one. All I needed to do was to create a list of lists or table with 3 rows and 3 columns. The way I created this table was the same when I create it in Java or C++, so I did not have any problem. Finally, the last step was to print the word from the table according to the rows and columns from the user. I subtracted it by 1 because the parameters work as an index. For example, if the user picks row 1, so the index for row 1 is 0.

Text

Description automatically generated

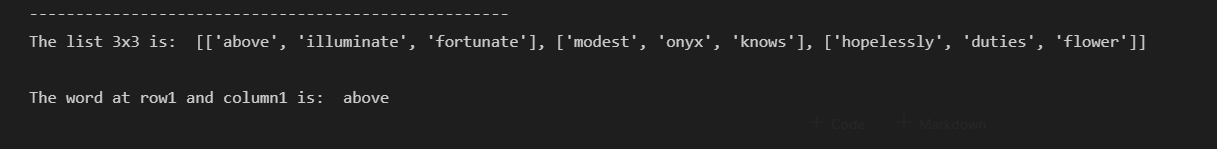
Overall, the assignment was fun to code and useful to sharp my programming skills.

**Part 1 Test 1**

**Text

Description automatically generated**

**Part 2 Test 1**

****