SE113 – LAB#9 2023-2024 SPRING

Aim: File handling, understanding lists and strings.

Preliminary work to do:

Download the following file as "txt" from web site for the words to be used in this Lab work (right-click the link and save as txt):

https://www.mit.edu/~ecprice/wordlist.10000

Read the text file as a whole into a variable called `many_words`.

Split `many_words` and store the resulting list into a variable called `many_words_list`.

1. In this lab you will write a Python script that will generate random passwords from the given list variable `many_words_list`. Your script should start by assigning a variable `user_number` to a value to be entered by the user. This number should be validated and must be between 3 and 7, inclusive. This value would refer to the number of words to be chosen randomly from the list variable `many_words_list`. Depending on the value of `user_number`, your code should randomly choose that number of words from the `many_words_list`. Lastly, your script should concatenate these words into one string to create the password and print it on the screen.

Hint: You will need "random" to be imported. You are free to explore its methods.

2. Next, define the following two functions in your code:

`rep_with_upper` - This function takes a string as its only parameter and returns a new string value. The function should create the new string value by replacing a character randomly chosen inside the given parameter with its uppercase version (e.g. a --> A). In your modified script, you will call this function by passing the password created in Question#1 as the argument value.

`swap_letters` - This function takes a string as its only parameter and returns a new string value. The function should create the new string value by swapping the first two characters of the given parameter with the ones in the last two (e.g. Password --> rdsswoPa). In your modified script, you will call this function by passing the password returned from `rep_with_upper` function as the argument value.

Now, modify your script for Question#1 in the following manner: Using your answer to Question#1, create the initial password; then using this password as the argument value to the `rep_with_upper` function create a new password; finally using this new password as the argument value to the `swap_letters` function create the final version of the password. Your modified script should print both the initial and the final versions of the password created.

SAMPLE OUTPUT (bold parts are entered by user):

```
Please enter a value (from 3 to 7) for the number of words: 13

Invalid value!

Please enter a value (from 3 to 7) for the number of words: 1

Invalid value!

Please enter a value (from 3 to 7) for the number of words: 4

Initial password: pythonsurelemonunix

Final version of the password: ixthonsuRelemonunpy
```

3. In this last question, add your code the following: Open a file named “store.txt” in write mode. Write your last password from the previous question. The file should have only one line, which is the password you have generated.

TODO@HOME

Modify your script for Question#2 by adding the following function: `search_letter`

`search_letter` - This function takes two strings and returns a bool value (i.e., either `True` or `False`). The first argument value will be the source string to be searched and the second argument value will be the key letter to be searched in the source. If the key letter is found in the source the function will return `True`, otherwise `False`.

Test your `search_letter` function in the following way: Call the function by passing the final version of the password created in Question#2 as the first argument value and your name’s first letter as the second argument value. Check the function’s return value to see the result.

Note: This last task is just an illustration of using many different functions in a script. Using such a function would not make sense in choosing strong passwords. Consider using different mechanisms to create strong passwords.