Chapter 6: Files and Exceptions

Password Program – Accounts

This assignment will create the ability to make accounts (the username and password), save them, and have a different program to allow login. If available, you may reuse the code from your previous program but must modify it to fit this project's parameters.

A series of valid usernames and passwords are to be stored in a text (.txt) file (for simplicity's sake, the file should be in the same directory as the python file). It will be formatted so that the first line is a username and the second is the username's password, the third line contains the second username and fourth its password, etc. This file will be created and appended to by the first program.

1. You will write a program (create_account.py) that simulates entering a new account into a computer. Every time this program is run, it will ask for a username and password to be added to its records. After the inputs are accepted, you will open a text file (account_details.txt) and add the records in the format specified in the above paragraph. You will run this program at least six times, entering a new record each iteration. Check the text file after each run to see if it renders correctly. If it does not, do not modify the records in the text file. You should rather fix the python file so it should render properly, delete the text file, and run again.

2. After create_account.py works properly, you will start the login_account.py file. This is the file that may reuse your previous code. After you ask for a username and password, you will open account_details.txt file and read in the records. Remember that username and login details come in two-line pairs. If the details match, display an enter message. Otherwise, display that the entries are incorrect and ask again.

Notes:

- The process for dealing with a file is:
    1. Open the file.
    2. Read contents; repeat as desired.
    3. Close the file.
- The default mode when opening a file is read. If you wish to open in a different format, you must specify the format. The three formats you need to know are read, write, and append. They each function differently.
- Whitespace characters can be removed by using the `string_variable.strip()` function. `strip` removes whitespace from both ends, `rstrip` removes from the right end, and you can guess the last version.
- You will encounter many bugs when writing this program. `print` statements are your friend. Run your program frequently to find errors soon after they are created.
- You can use a loop to read the lines of a program. They will be in two-line pairs. To prevent reading too far, you can test that a read line does not equal an empty string.