Protected Passwords

LAB # 1

By

Corey Henry

***“On my honor, as a Mississippi State University student, I have neither***

***given nor received unauthorized assistance on this academic work.”***

Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

CSE-1384-06-201430 Intermediate Computer Programming

Class Section # 6

Jesse Farek

1/26/3015

**Analysis and Conclusions**

This was a good exercise to get the cobwebs out and to get back into programming. You needed to know a lot about files and loops to get this done. The point was to be able to create or open a file that was password protected, and if you had the password display the information that It was given. I ran into a problem with my count controlled loop but ended up finding out it was a typo that was not getting caught due to the + and + being mixed up. Line slicing was also very important to be able to read one line to get the password out. Functions were important as well to set up the body of the code. If I could rewrite this code I would make the options format better and less repeats. It could be better written so that the options menu doesn’t show up as much, and I believe it could be written with less lines of code.

Source Code:

Corey Henry #Date Assigned: 05Nov14

# #

#Course CSE 1384 Sec 06 #Date Due: Jan 27

#File name: Lab1-protected files

#

#Program description- create and/or open a file that is password protected.

#create a function that will create a file

def create\_file():

#determine password and check to see if the equal each other

password = input('Please enter the password for the file you wish to create: ')

re\_pass = input('Please Re-enter the pass word for the file you wish to create: ')

while password != re\_pass:

print('Passwords do not match please try again')

password = input('Please enter the password for the file you wish to create: ')

re\_pass = input('Please Re-enter the pass word for the file you wish to create: ')

file\_name = input('Please enter the name of the file you wish to create: ')

#add .txt to the file

if not file\_name.endswith('.txt'):

file\_name += '.txt'

#open file and pass the password and any information they want too and set it up to cancel when they enter :wq

#dont add :wq to the file

infile = open(file\_name, 'w')

infile.write(password)

information = '\n'

while information != ':wq':

information = input('Please enter the information you wish to add to the file: ')

infile.write('\n')

if information != ':wq':

infile.write(information)

infile.close()

#function to open the file and read the password.

def open\_file():

infile = input('what is the file name you wish to open? ')

if not infile.endswith('.txt'):

infile += ('.txt')

count = 1

infile = open(infile, 'r')

password = infile.readline()

password= password.strip('\n')

information= infile.read()

infile.close()

prompt = 'wrong'

#while loop to give 3 chances to get password correct and print the information

while prompt != password and count < 4:

prompt = input('Please enter the password to open the file: ')

count +=1

if prompt == password:

print(information)

else:

return

#main function to give them the option of opening or creating the file

def main():

option = input('Would you like to open or create a file(type stop to end program)? ')

if option == 'open':

open\_file()

main()

if option == 'create':

create\_file()

main()

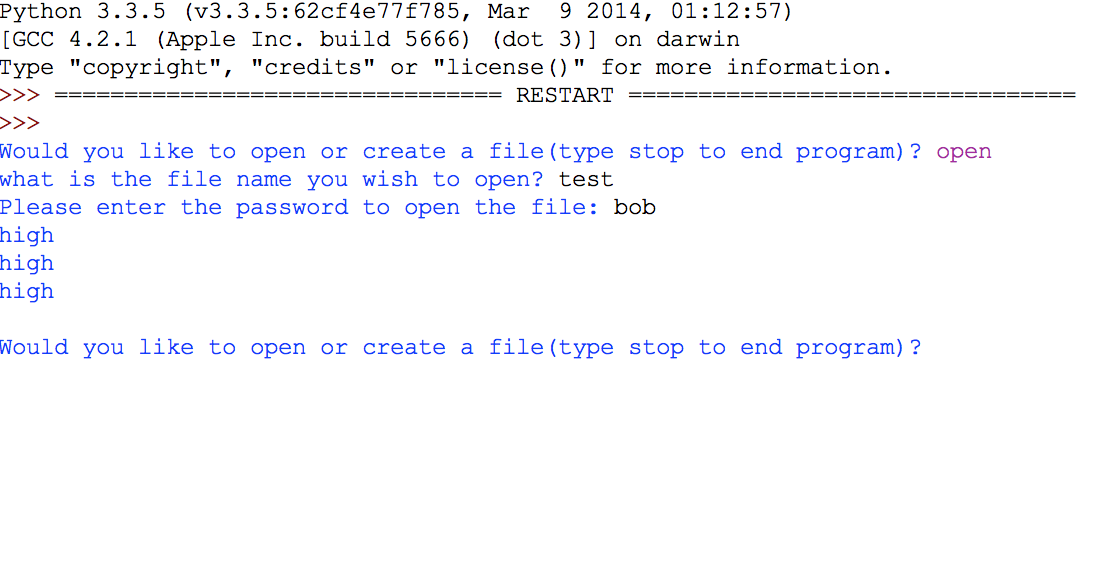
else:

quit

# run main

main()

Example:



|  |  |  |
| --- | --- | --- |
| Test Case Number | Action | Expected output |
| 1 | Choose create or open file | If open, asks for file name, if create asks you for the password for the new file |
| 2 | Create a password | Asks you to enter a password, then re-renter password, if both match allows you to continue |
| 3 | Asks you for file name | Requests you to type in a name for the file |
| 4 | Tells you to type in the information, when finished you type in ‘wq’ | Should take all information written and should not end until :wq is typed in, but this should not be written to the file |
| 5 | Open a file | Asks you for the file name and the password, if password is incorrect 3 times, program starts over. |
| 6 | Display information | Should display information from file minus the password and minus the :wq that ended the information portion of create a file. |

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case Number | Date and Time | Actual Output | Pass or Fail |
| 1 | Jan 26 16:29 | Enter password | Pass |
| 2 | Jan 26 16:29 | Passwords match continues | Pass |
| 2 | Jan 26 16:29 | Passwords do not match try again | Pass |
| 3 | Jan 26 16:29 | Continues after named | Pass |
| 4 | Jan 26 16:29 | Accepts information until :wq | Pass |
| 5 | Jan 26 16:29 | Password works | Pass |
| 6 | Jan 26 16:29 | Displays file information without password or :wq | Pass |