

```
#!/usr/bin/env python
# coding: utf-8

# In[13]:

'''
(1) you read a text file programmatically that it has colons ( : ) as the
delimiter between its data elements.
(2) pick specific items in the file and write
the result into a .csv file.
(3) read that .csv file and show the result on the screen .
I have provided a text file in the Blackboard.
That is a sample Linux password
file. The name of the file is the password.txt file.
(4) Use that password.csv to test your code.
Please consider that if there is no missing data, each row has 10 data elements, separated
by colons like:

nobody:*:-2:-2::0:0:Unprivileged User:/var/empty:/usr/bin/false

We are interested to know about the users that have shell access (hence there is no /false
in their corresponding line).

(5) In the output csv file, we will have just two columns, the
user name and the shell program they use (like sh , csh, etc).
(6) Make sure your code filters (ignores) the lines if they do not
have "both" of user name and shell (i.e we are only
interested to show the lines that have both the user name and the shell. If only one of
them exists, we ignore that line altogether, like the instance 1 above). Also, make sure
you do not include the lines that do not have all the 10 elements or their placeholders
(like instance 2 above). So you do not have to include the "orange" user in the output,
in the same data set. The code at the end needs to read the output .csv file to show the
result in the notebookSample'''

import os
import pandas as pd
f = pd.read_csv('password.txt')
# f.columns = ["username", "shell program"]

#exclude entries that do not have both user name and shell - first and last delimited part of password

usernames = []
shells = []

#delimit by colon

for line in open('password.txt'):
    l = line.strip().split(':')
    #     print(l[0], l[-1]);
    username = l[0]
    shell = l[-1]
# skip strings that do not have all 10 data elements
    if len(l) != 10:
        continue
# Skips false shells and empty username and shells
    if shell.find('false') > -1 or username == '' or shell == '':
        continue
    usernames.append(username)
    shells.append(shell)
#write into data frame with pandas
df = pd.DataFrame({'username': usernames, 'shell': shells})

#generate csv file
directory = "Password Report"
parent_dir = "../Code Along"
path = os.path.join(parent_dir, directory)
if not os.path.exists(path):
    os.mkdir(path)

    if not os.path.exists(path):
        f = open(path, "x")
    f=open(path, 'w')

df.to_csv('Password Report/report.csv')
df.head()

# In[14]:

'''In this question, you write a code that creates a sub-folder in the current folder and it
generates a random number of files, but not less than 10 files (at least 10 files). In each file,
the code picks 20 random words from the dictionary file that is uploaded into the Blackboard.
Each word must be in one line. Your code will search through all files and finds the longest
word in each file and show a
```

report like the name of the file and its longest word:
File_name: the longest wordSample output:''

```
import os
import random

#create subfolder in parent directory

directory = "PRO675S1M"
parent_dir = "../Code Along"
path = os.path.join(parent_dir, directory)
if not os.path.exists(path):
    os.mkdir(path)
print("Directory '%s' created" %directory)

number_of_files = random.randrange(10, 15)
print(f'{number_of_files} files to be printed')

for i in range(number_of_files):
    # select 20 random words from the dictionary
    random_words = []
    with open("dictionary.txt", 'r+') as f:
        words = f.read().split()
        random_words = random.choices(words, k=20)
    # create the file
    file_name = "myfile"+str(i)+ ".txt"
    file_path = os.path.join(path, file_name)
    if not os.path.exists(file_path):
        f = open(file_path, "x")
    # write the 20 random words to the file

    f=open(file_path, 'w')

    for word in random_words:
        # print(word)
        f.write(word+"\n")
    f.close()
    # file.write(random_words + '\n')

#create function that checks for the longest word

def longestword():
    directory = "PRO675S1M"
    parent_dir = "../Code Along"
    path = os.path.join(parent_dir, directory)
    for i in range(number_of_files):
        file_name = "myfile"+str(i)+ ".txt"
        file_path = os.path.join(path, file_name)
        with open(file_path, 'r+') as f:
            words = f.read().split()
            max_len = 0
            max_word = ''
            for word in words:
                if len(word) > max_len:
                    max_len = len(word)
                    max_word = word
            print(file_name , ' ', max_word)

longestword()
```