```
"""# -*- coding: utf-8 -*-
Created on Sat Dec 3 07:30:14 2022
@author: 17574, b.hogan@snhu.edu
it.304 wk14 Project Part II
                         =====
#
                      =====
#
                                =====
#
                =====
                                   =====
                >_7 Pillars of Python Project
#
                =====
#
#
                      =====
                            =====
                         =====
# 1) Perform with ANYONE!
# 2) Copy answers from CONSOLE, put under question, and comment out
      use a pound sign # for a single line (pound sign is the proper name)
      Comment multiple lines like
""" <== START
   your answers here
""" #end
# 3) Email me you current project work however it is ! Please do so now.
# 4) Prep and check the image of what the data should like in the directory
# 5) PLEASE read the instructions.
                             May the grok be with you ~brian
                       ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
#=> Part 0.0 - FILL IN THE BLANKS
#=>
                            LOSE 20 points if not 100% correct!
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
                       ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
.. .. ..
    obj Name
             l charcter code
                             | explicit code
                             | -----
    -----
                             |=> mytuple =
 i) mytuple = |
ii) mylist =
                             |=> mylist =
                            |=> mydict =
iii) mydict = |
iv) myset = |
                           => myset =
 v) dataframe =
                            |=> df =
                            |=> mystring=
vi) mystring = |
#______
#=> Part 0.1 - Download files
#-----
# a) CREATE a folder in your data folder called Shakespear_txt_name
# b) unpack the zipped github shakespeare corpus text files into this folder.
    https://github.com/bbe2/IT.304.Fall.2022/blob/Shakespeare-Corpus/shakespeare_txt_fullname.zip
# c) use your SECOND monitor and verify with this image
# d) if you dont have a second monitor please let me know so I can get you one
# https://github.com/bbe2/IT.304.Fall.2022/blob/main/xwk_14_Project_Text_File_Validation_Step_0.pdf
```

```
~ ~ ~ ~ ~ ~ ~ ~ ~ ~
                            ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
#=> Part 1: IMPORT DATA (done for you)
#
# INSTRUCTIONS
# a) change path to match your directory.
# b) it should ONLY involved deleting " it304" in the path below
# c) Please text me if not done in 5 min
                       #.7-5-7.four.Se~@Ven.7.8!~>2>?4=1
import pandas as pd
                                                 #dataframe library
#df = pd.read_excel("."),df.to_dict(),pd.DataFrame.from_dict(mydict)
import numpy as np
                                                #num library
import matplotlib.pyplot as plt
                                                #visualization library
                                                 #op system, #msft.os=\\
import os
import sys
                                                 # sys.exit()
# inform the operating system what drive and location working on
os.chdir('C:\\Users\\17574\\Desktop\\data it304\\shakespeare txt name')
# create a path so you can use in code to go find files in a directory
path = 'C:\\Users\\17574\\Desktop\\data_it304\\shakespeare_txt_name'
mylist_filenames = os.listdir(path)
help(os.listdir)
                   #=> help(os.listdir)
                   #=> Help on built-in function listdir in module nt:
                   #=> Return a list containing names of the files in directory
print(mylist filenames)
print(type(mylist filenames)) #class list
print(len(mylist_filenames))
#=> Part 1 cont: INSTRUCTIONS
#=> double check your import for mylist_filenames equals answer 4 lines down
#=> EXTRA POINTS if double check programmatically, hint: myA == myB
#=>ANSWER
# mylist_filenames should match this list
    ['A Midsummer Nights Dream.txt', 'Alls Well That Ends Well.txt',
     'Antony and Cleopatra.txt', 'As You Like It.txt', 'Comedy of Errors.txt',
     'Cymbeline.txt', 'Hamlet.txt', 'Henry IV part 1.txt', 'Henry IV part 2.txt',
     'Henry V.txt', 'Henry VI part 1.txt', 'Henry VI part 2.txt',
     'Henry VI part 3.txt', 'Henry VIII.txt', 'King Lear.txt',
     'Loves Labours Lost.txt', 'Macbeth.txt', 'Measure for Measure.txt',
     'Much Ado About Nothing.txt', 'Othello the Moore of Venice.txt',
     'Pericles.txt', 'Richard II.txt', 'Richard III.txt', 'Romeo and Juliet.txt',
     'The Life and Death of Julius Caesar.txt',
     'The Life and Death of King John.txt', 'The Merchant of Venice.txt',
     'The Merry Wives of Windsor.txt', 'The Taming of the Shrew.txt',
     'The Tempest.txt', 'The Tragedy of Coriolanus.txt', 'Timon of Athens.txt', 'Titus Andronicus.txt', 'Troilus and Cressida.txt', 'Twelfth Night.txt',
     'Two Gentlemen of Verona .txt', 'Winters Tale.txt']
    # <class 'list'>
    # Out[3]: 37
```

~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~

```
#=> Part 1 cont: INSTRUCTIONS
#=> the following seperates file name from its extension
#=> PLEASE double check correct. ifnot.text.ME.7-5-7.four.Se~@Ven.7.8!~>2>?4=1
#-----
mylist_playnames= []
for file in os.listdir(path):
    #print(path+ "\\" + file)
    \#next = path + "\" + file
    filename = file.split(".") #=> get names and file paths in any directory
                                #=> returns list and title in index 0
    justname = filename[0]
    print(justname)
    mylist_playnames.append(justname)
print(len(mylist_playnames))
#=>ANSWER
.. .. ..
        A Midsummer Nights Dream
        Alls Well That Ends Well
        Antony and Cleopatra
        As You Like It
        Comedy of Errors
        Cymbeline
        Hamlet
        Henry IV part 1
        Henry IV part 2
        Henry V
        Henry VI part 1
        Henry VI part 2
        Henry VI part 3
        Henry VIII
        King Lear
        Loves Labours Lost
        Macbeth
        Measure for Measure
        Much Ado About Nothing
        Othello the Moore of Venice
        Pericles
        Richard II
        Richard III
        Romeo and Juliet
        The Life and Death of Julius Caesar
        The Life and Death of King John
        The Merchant of Venice
        The Merry Wives of Windsor
        The Taming of the Shrew
        The Tempest
        The Tragedy of Coriolanus
        Timon of Athens
        Titus Andronicus
        Troilus and Cressida
        Twelfth Night
        Two Gentlemen of Verona
        Winters Tale
print(filename)
                        #['Winters Tale', 'txt']
print(justname)
                        # Winters Tale
print(type(filename))
                       #<class 'list'>
print(path+ "\\" + file)
    #C:\Users\17574\Desktop\data_it304\shakespeare_txt_name\Winters               Tale.txt
#=> INSTRUCTIONS: copy and paste here your path
```

```
#=> urANSWER
```

```
#=> Part 1 cont: INSTRUCTIONS
#=> View the variable explorer. Do you have more than 2 names in it?
#=> List ALL the files you have here
#=> urANSWER
#
#-----
#perform housekeeping
del file; del filename; del justname
print(type(mylist_playnames))  # <class 'list'>
print(type(mylist_playnames[0]))  # <class 'str'>
```

```
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
                       ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
    ~~~~~~~~
#
#=> Part 2: LEARN to count words and characters
#=> BUILD your skills looping and counting words and characters
~ ~ ~ ~ ~ ~ ~ ~ ~ ~
#=> BEHAVIORS: if data packed as a string you can count characters
            if data packed as a list you are counting words
#=>
#=>
            cast data from lists to strings or strings to lists
#=>
            depending on what and how you want to count
            you may need to use iterators to do this!
#=> Part 2 cont: COUNTing LISTS vs WORDS vs CHARACTERS
#-----
#=> a space or carriage return between one word and another is a "character"
print(len("ab")) # 2
print(len("a bc"))
                 # 4
print(len("a b c")) # 5
print(type(mylist playnames[0]))
                                   # <class 'str'>
print(mylist_playnames[0])
                                   # A Midsummer Nights Dream
print(mylist_playnames[1])
                                    # Alls Well That Ends Well
print(mylist_playnames[2])
                                   # Antony and Cleopatra
print(mylist playnames[3])
                                    # As You Like It
                                   # 4, count at word level
print(len(mylist_playnames[0].split()))
print(len(mylist_playnames[1].split()))
                                    # 5, count at word level
print(len(mylist_playnames[2].split()))
                                    # 3, count at word level
print(len(mylist_playnames[3].split()))
                                    # 4, count at word level
#----COUNT AT ITEM LEVEL
count 1 = 0
for i in mylist_playnames: # count at item level, not word or character
   count_1 +=1
               #37 titles in the list
print(count_1)
#-----COUNT AT CHARACTER LEVEL
count 2 = 0
for i in mylist_playnames[0]:
   count 2 +=1
                      # 24 characters in "A Midsummer Nights Dream"
print(count_2)
print(len(mylist_playnames[0].split())) # BUT the length of split words is 4
#----- everything by words or characters
#=> this counts the total words for each script title and appends to a new list
mylist words title = []
for i in mylist_playnames:
   mylist_words_title.append(len(i.split()))
print(mylist_words_title)
   # [4, 5, 3, 4, 3, 1, 1, 4, 4, 2, 4, 4, 4, 2, 2, 3, 1, 3, 4, 5, 1, 2, 2,
   # 3, 7, 7, 4, 5, 5, 2, 4, 3, 2, 3, 2, 4, 2]
print(min(mylist_words_title), max(mylist_words_title), sum(mylist_words_title))
'''#=> 'ANSWER: min, max, total = 1, 7, 121'''
         b.hogan@snhu.edu, it.304.wk.14.project.II
```

```
~ ~ ~ ~ ~ ~ ~ ~ ~ ~
     ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
                          ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
#=> Part 3: BUILD A SCRIPT MEGASARUS (done for you)
#-----
     ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
mylist_script_megasaurus= []
for file in os.listdir(path):
                                # 'r' parameter to read vs 'w' to write
   with open(file,'r') as data: # object.readlines is a method
       mylist_script_megasaurus.append(data.readlines())
#help(data.readlines)
                                       # use help to learn unknown methods
#print(mylist_script_megasaurus)
                                       # HUGE !
print(len(mylist script megasaurus))
                                       # 37 scripts in one list!
print(type(mylist_script_megasaurus))
                                      # <class 'list'> titles in a list
#~~~~~~~ CRTICAL PACK AND UNPACK POINT
# -----the data is packed as a list, not a string
# -----everything you did up top is with a string
# .....AH Soooooooooo
print(type(mylist_script_megasaurus[0])) # <class 'list'>
print(len(mylist script megasaurus[0]))
                                     # 1, only 1 script in item 1
#=> INSTRUCTION
# Add ONE function to this statement to count the script as a string
# You can use any combination of iterators, etc to do counting
# I WANT TO SEE ALL COMBINATIONS RIGHT OR WRONG
# Please type every combination right here spending 30-60 minutes max
# Answers -
#
          words
                     characters
#
          16,026
                     81,505
# HINT --> function A(function B(object).method<functionB>())
len(mylist script megasaurus[0])
#=> could also try
for i in mylist_script_megasaurus[0]:
   True #true for compiling only would delete if you use
# urANSWER for Part 3
#===> the answer
len(str(mylist script megasaurus[0]))
                                           #81505
len(str(mylist_script_megasaurus[0]).split())
                                           #16026
```

```
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
                          ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
#=> Part 4: REPEAT FOR THE WHOLE CORPUS (all the scripts)
#=> Again, I WANT TO SEE ALL COMBINATIONS RIGHT OR WRONG
mylist words= []
mylist characters= []
for i in mylist_script_megasaurus:
   for word in i:
       mylist_words.append(len(str(word).split()))
       mylist_characters.append(len(str(word)))
print(mylist words)
   # [ 16026, 22527, 71154, 42565, 14495, 26653, 29479, 23874, 25726, 25513,
       20507, 24399, 23279, 23148, 25238, 21051, 16385, 21228, 20749, 25750,
       17422, 21740, 28285, 23743, 19052, 20349, 20899, 21041, 20374, 16003,
       26455, 17366, 19782, 25222, 19395, 16857, 24471]
print(mylist characters)
   # [ 81501, 113219, 364289, 211738, 72787, 136054, 149870, 120676, 131720,
       132051, 108775, 126099, 120398, 118465, 129344, 107350, 84811, 107557,
       102728, 129652, 88879, 112713, 145669, 119313, 96398, 104880, 104502,
       105018, 101852, 81695, 135205, 89819, 101909, 129526, 95784, 84526,
       124131]
print(min(mylist_words), max(mylist_words), sum(mylist_words))
print(min(mylist_characters), max(mylist_characters), sum(mylist_characters))
              Min
                      Max
                             Total
# Words
              14,495 71,154 878,202
# Characters
              72,787 364,289 4,470,903
```

```
~ ~ ~ ~ ~ ~ ~ ~ ~ ~
                            ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
#=> Part 5: Explain to me everything going on below
~ ~ ~ ~ ~ ~ ~ ~ ~ ~
     ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
#urANSWER
.....
.....
mylist_id = []
mylist id = list(range(37))
print(type(mylist_id))
print(mylist id)
   # <class 'list'>
   # [ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19,
       20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36]
comedy = ['A Midsummer Nights Dream', 'Alls Well That Ends Well',
'As You Like It',
 'Comedy of Errors', 'Cymbeline', 'Hamlet',
'Loves Labours Lost',
 'Measure for Measure', 'Much Ado About Nothing',
 'Pericles',
 'The Merchant of Venice', 'The Merry Wives of Windsor', 'The Taming of the Shrew', 'The Tempest',
 'Twelfth Night', 'Two Gentlemen of Verona ', 'Winters Tale']
history = [ 'Henry IV part 1', 'Henry IV part 2',
 'Henry V', 'Henry VI part 1', 'Henry VI part 2', 'Henry VI part 3',
 'Henry VIII', 'Richard II', 'Richard III', 'The Life and Death of King John']
tragedy = ['Antony and Cleopatra', 'The Tragedy of Coriolanus', 'Macbeth',
 'Timon of Athens', 'Titus Andronicus', 'Troilus and Cressida',
 'Othello the Moore of Venice', 'Romeo and Juliet',
 'The Life and Death of Julius Caesar', 'King Lear', ]
print(len(comedy), len(history),len(tragedy))
#17 10 10
mylist_play_type = []
for i in mylist_playnames:
   if i in comedy: mylist_play_type.append("comedy")
   if i in history: mylist play type.append("history")
   if i in tragedy: mylist play type.append("tragedy")
print(mylist_play_type)
print(len(mylist play type))
# 1f) create a dictionary that matchs weeks 5-8 input spreadsheet
    => title, script, type, id, words_script, words_title
mydict = {}
mydict = {'title':mylist_playnames, 'script':mylist_script_megasaurus,
          'type':mylist play type, 'id':mylist id,
          'words_script':mylist_words, 'words_title':mylist_words_title}
```

```
#=> 1g) send dict to df, export to spreadsheet, email to me
#-----
df1 = pd.DataFrame.from_dict(mydict)
df1.info()
df1
#make sure you put in a different directory
os.chdir('C:\\Users\\17574\\Desktop\\data it304')
mywriter = pd.ExcelWriter('my_wk14_Project_Summary_Report.xlsx')
df1.to excel(mywriter)
mywriter.save()
#=> Part 5 continue - create summary report by play type
Total all script words and title words by 3 play types
 #
         send to df to spreadsheet and email to me
 #
       Answer:
         # comedy + history + tragedy
         # 371235 + 236820 + 270147 = 878202
                + 35
                            + 33
                                       = 121
comedy script words = 0; history script words = 0; tragedy script words = 0
comedy_title_words = 0; history_title_words = 0; tragedy_title_words = 0
i=0
while i <=36:
   if mylist_playnames[i] in comedy:
       comedy_script_words = comedy_script_words + mylist_words[i]
       comedy title words = comedy title words + mylist words title[i]
   if mylist_playnames[i] in history:
       history_script_words = history_script_words + mylist_words[i]
       history_title_words = history_title_words + mylist_words_title[i]
   if mylist_playnames[i] in tragedy:
       tragedy_script_words = tragedy_script_words + mylist_words[i]
       tragedy title words = tragedy title words + mylist words title[i]
   i = i+1
mydict2 = {"comedy_script_words":comedy_script_words,
          "history_script_words":history_script_words,
          "tragedy script words":tragedy script words,
          "comedy_title_words":comedy_title_words,
          "history_title_words":history_title_words,
          "tragedy_title_words":tragedy_title_words}
mydict2
   # {'comedy script words': 371235,
   # 'history_script_words': 236820,
   # 'tragedy_script_words': 270147,
   # 'comedy title words': 53,
   # 'history_title_words': 35,
   # 'tragedy_title_words': 33}
df2 = pd.DataFrame.from_dict(mydict2, orient='index')
df2.info()
mywriter = pd.ExcelWriter('my_wk14_Project_total_words.xlsx')
df2.to excel(mywriter)
mywriter.save()
```

#=> Future Team class work #-----Create an object with one or two functions. Ask user what play they want to read. Figure out a minimum of 1 other useful piece of information # to display or include in user report. Have function export data # you can simply email me the file!