LAB 05

SUBMISSION INSTRUCTIONS

Submit 1 python file using the naming convention below (replace JaneDoe with your first and last name respectively):

- JaneDoe5.py

QUESTION

1. Create a folder named files then download and store input.txt in it.
2. Write program that will:
   a. Ask the user for the input file name (use exceptions to deal with a missing file).
   b. Once the right file name is provided, create an output file named output.csv.
      i. This file should also be stored in the files folder.
      ii. Please note, before you store output.csv in the files folder, you need to check if there already exists a file with that name in the folder. Use the os.path module (not exception handling) to check if the file already exists. If it does, ask the user if they want to overwrite it or if they would prefer creating a different output file (refer to sample output for the logic). Regarding the os module, the code below is enough to get you started.

      ```python
      import os.path
      if os.path.isfile("Presidents.txt"):
          print("Presidents.txt exists")
      ```
      Here isfile("Presidents.txt") returns True if the file Presidents.txt exists in the current directory.

      iii. If no file exists with the name output.csv, a new file with that name should be created. If the file already exists, ask the user if they want to overwrite it. If they select y for yes, you can use the w file mode to overwrite the file. If they select n for no, ask them for a new output file name.
   c. Once the output.csv file is successfully created, create a header row in it with the headers: Email, Time, and Confidence.
   d. Go through the lines in input.txt looking for lines that begin with (please note the colons are part of the text)

      ```
      To: source@collab.sakaiproject.org
      From: stephen.marquard@uct.ac.za
      Subject: [sakai] svn commit: r39772 - content/branches/sakai_2-5-x/content-impl/impl/src/java/org/sakaiproject
      X-Content-Type-Outer-Envelope: text/plain; charset=UTF-8
      X-Content-Type-Message-Body: text/plain; charset=UTF-8
      Content-Type: text/plain; charset=UTF-8
      X-DSPAM-Result: Innocent
      X-DSPAM-Processed: Sat Jan  5 09:14:16 2008
      X-DSPAM-Confidence: 0.8475
      X-DSPAM-Probability: 0.0000
      ```

      i. From: - Extract the email address (e.g., stephen.marquard@uct.ac.za)
      ii. X-DSPAM-Processed: - Extract the time value (e.g., 09:14:16)
      iii. X-DSPAM-Confidence: - Extract the numeric value (e.g., 0.8475)
   e. Send the results obtained in (d) to output.csv for each occurrence of the 3 items in input.txt.
   f. The contents of your output.csv file should be identical to sample.csv when your program is done (please note the last row contains the average X-DSPAM-Confidence). If no values are found, which would cause an issue when averaging, you should raise a ZeroDivisionError from the function that you are doing the

logic in, catching the exception with a try/except block in the code that calls that function (as discussed in class).

g. Once all the data is sent to output.csv, print Data stored! to the console to let users know your program is done.

Please note:

- Write your program as a script (i.e., include the if __name__ == '__main__': block).
- Strip input when getting the file name (do not lower). There are some OS systems that are case sensitive with file names so make sure you don't change the case of the file name entered (only strip file name input).
- Your program should be able to work if the input and output file names were different.
- Don't forget to close any files you open.
- Feel free to use functions to organize your code.

SAMPLE OUTPUT 1

Input file name: input.txt
Output file name: output.csv
Data stored!
NOTE: The assumption in this scenario is that you didn't have output.csv in your files folder so this is the first time it's being created.

SAMPLE OUTPUT 2

Input file name: input.txt
Output file name: output.csv
Overwrite existing file (y/n): Y
Data stored!
NOTE: The assumption in this scenario is that you already had a file named output.csv in your files folder but you are choosing to delete its previous content and send new data to it.

SAMPLE OUTPUT 3

Input file name: input.txt
Output file name: output.csv
No Confidence values present!
NOTE: The assumption in this scenario is that you didn't have output.csv in your files folder so this is the first time it's being created. Additionally the assumption is that the input file does not contain any of the values that the program is looking for.

SAMPLE OUTPUT 4

Input file name: test.txt
File does not exist!
Input file name: input
File does not exist!
Input file name: input.txt
Output file name: output.csv
Overwrite existing file (y/n): x
Enter (y/n): a

Enter (y/n):   N
New output file name: output.csv
Overwrite existing file (y/n): n
New output file name: test.csv
Data stored!
NOTE: The assumption in this scenario is that you already had a file named output.csv in your files folder and are choosing to send your data to a new csv file named test.csv because you don't want to overwrite your output.csv file.