Chapter 7 & 8 assignment will be a group coding competition. Write one script that include multiple functions to read an input file that outputs both valid and invalid data files.

When working in or outside the classroom, you should be working as a group. Which means, when I review each group's GitHub repositories, I expect to see that the person checking in the code is regularly alternating from one person to the other.

Groups will present their completed code and output files to the entire class on Thursday, April 14.  Each person will get one vote on the best solutions.

Make up your own input file that will fully test your program's logic.  Example:

1|Johnson,Debbie|dejohns2@wsc.edu|111-222-3333  
1|Johnson|dejohns2@gmail.com|1112223333  
abc|Johnson,Debbie|dejohns2@wsc.edu|111-222-3333  
Johnson,Debbie|dejohns2@wsc.edu|111-222-3333

Hints:

* Read the data one line at a time
* Do NOT load the data into a list, instead immediately write the data to the valid or invalid file one line at a time
* Use meaningful file names, example: input\_data.csv, valid\_data.csv, invalid\_data.csv
* Use constants for your file names, example: INPUT\_FILENAME = 'chapter7\_8\_Competition\input\_file.csv'
* Use meaningful file object variable names, examples: input\_file, valid\_file, invalid\_file
* Displays errors if you are unable to open a file, read the entire file, or write to a file
  + Chapter 8 slide 16
* Use a finally clause to display how many valid and invalid records were processed
  + Chapter 8 slide 26
* Use the newline='' named param for opening the file and Use the CSV module with delimiter set to '|'
  + Chapter 7's slide 21, 22
* Use a try-except when unpacking the row into variables in case there are too few items, example:

for row in reader:  
 try:  
 id, name, email, phone = row

input\_data.csv:

* **Example of valid data:** 1|Johnson,Debbie|dejohns2@wsc.edu|111-222-3333
* **Examples of invalid data:**   
  abc|Johnson|dejohns2@gmail.com|1112223333  
  Johnson|dejohns2@gmail.com|111-222-3333

valid\_data.csv

* Comma delimited
* Store first name and last name as separate fields
* Store phone numbers as 111.222.3333 format
* **Example of valid data:** 1,Debbie, Johnson,[dejohns2@wsc.edu,](mailto:dejohns2@wsc.edu,)111.222.3333

invalid\_data.csv

* Will contain an error code plus the input data
* Invalid record length (if so, you will NOT check any other data)
* Invalid id that isn't an integer
* Invalid names that are not in a "last name, first name" format
* Invalid email that are not in proper email format or not .edu extension
* Invalid phone numbers that are not in a 111-222-3333 format
* **Example of invalid data:**   
  L|Johnson|dejohns2@gmail.com|1112223333  
  INE|abc|Johnson|dejohns2@gmail.com|111-222-3333
  + L = invalid record length (no other errors will also display)
  + I = invalid id
  + N = invalid name
  + E = invalid email
  + P = invalid phone

Make sure to include your GitHub URL in your script level docstring

Attach your Python script, all input and output files, plus the console text file.  Use the comment area for your assignment reflection.