Scripting and Computer Environments - CSE 505
IIIT Hyderabad - Monsoon 2016
LAB WORK

Lab No. 7
Lab Date: 1/11/2016

Guidelines :

1. Read all questions carefully.
2. Please use *man* and *apropos* before calling a TA for help.

Submission Format :

1. Create a folder <rollno>_lab<lab_no>. All your work should be in this folderONLY
2. You have to generate a script file with the name <rollno>_ lab<lab no>.script
3. Create <Question_no>.py for all questions in section
4. Directory Structure :
   - |d- <rollno>_lab<lab_no>
   - |f- <Question_no>.py
   - |d/f- All files/directories created in this session
5. Zip the folder to <rollno>_lab<lab_no>.tar and upload it

Lab Questions

1. You are given *Marks.csv* file which contains marks assigned to each student on a scale of 10. Now your task is to generate *Grades-final.csv* file which will hold the <student-name, Total marks obtained by the student in all the labs & assignments, Average of the total, Grade>. The grade for each student is computed based on the conditions given in the below table.

Condition	**Normalized Grade**
Total marks >= Average(All students marks)	B
Total marks < Average(All students marks)	C

After performing the above operations, the intermediate output(NOT the final output)

would be(Here the average(All students marks) is 50.5)

```
prabhakar@prabhakar-Vostro-3550:~/temp
$ cat Grades-PreFinal.csv
Name, Total for all Labs and Assignments, Average of all labs & assignments, Grade
Amar, 49, 8.167, C
Akbar, 49, 8.167, C
Anthony, 51, 8.5, B
Gonzalviz, 50, 8.333, C
Bajji rao, 52, 8.667, B
Mastani, 52, 8.667, B
```

Now in the intermediate output, based on the *total-marks* the top 2 students should be given a A grade while the bottom 2 students should be given D grade. The final output should be similar to the one given in below image.

```
prabhakar@prabhakar-Vostro-3550:~/temp
$ cat Grades-Final.csv
Name, Total for all Labs and Assignments, Average of all labs & assignments, Grade
Amar, 49, 8.167, D
Akbar, 49, 8.167, D
Anthony, 51, 8.5, B
Gonzalviz, 50, 8.333, C
Bajji rao, 52, 8.667, A
Mastani, 52, 8.667, A
```

Store your Final output in *Grades-Final.csv.*
Note: You are **NOT** supposed to store intermediate results in any temporary files
How to take input?

$ python *Q1.py Marks.csv* ### Output should get saved in *Grades-Final.csv*

2. Its an extension of Q1. You are expected to pass the list of lines read from Marks.txt to the function getReportFromMarks() defined in the Q1.py and the function q1.getReportFromMarks() returns you the list containing the graded marks.

```
In [18]: gradedLines = q1.getReportFromMarks(lines)

In [19]: print gradedLines
[['Amar', '49', '8.167', 'D'], ['Akbar', '49', '8.167', 'D'], ['Anthony', '51',
'8.5', 'B'], ['Gonzalviz', '50', '8.333', 'C'], ['Bajji rao', '52', '8.667', 'A'
], ['Mastani', '52', '8.667', 'A']]
```

3. Write program
    a. Source1.py -> having DivideByZero Exception
    b. Source2.py -> having TypeError Exception
    c. Source3.py -> having Assertion error
    d. Source4.py -> having ImportError
    e. Source5.py ->having KeyError

And handle these exceptions : print the type of exception occured

4. Given a list of urls from user's browsing history in file "urls.txt", report the website he visits most frequently. Cases to be handled :
   a . If the url pattern is such :
      http://www.google.co.in/12345
      http://www.google.co.in/abcdefg
      Then these two are considered to be same website, i.e www.google.co.in
   b. Maintain the frequency of access of all urls
   c. Report most frequently visited URLS
   d. In case of tie, print all URLs with same frequency in sorted order. Each URL in a newline.

5. You are given a dataset containing some 'M' rows and 'N' columns. Values can be, binary(YES / NO, 0 / 1), continuous, discrete, categorical(Eg : LOW / HIGH / MEDIUM), NA(Not Applicable). You work is to replace all "NA" with some appropriate value that doesn't disturb the dataset much.
   Rules to replace NA:
   a. In numeric valued columns replace NA with the average of that column.
   b. In columns where non-numeric data is present, replace the NA with most frequent entry in that column

   Input FIle: dataset.csv
   Output FIle: new_dataset.csv
   NOTE:
   Make your program generic, we will check it with other dataset as well, or we will change the column order, or we will change the values of columns

6. Question: (S)
   You can find 4 HTML files page1.html, page2.html, page3.html, page4.html.
   These files will be present in a folder called HTMLPages
   All these files contain various HTML tags.
   We need you to focus on the anchor tag.
   Create a map (or anything you wish) of all the different/unique anchor tags(or links) along with the number of times they are present across all the 4 html pages.

ie: if we have page1.html : 1 anchor tag referring to a html page. 1.html
1 anchor tag referring to a html page. 3.html
page2.html : 1 anchor tag referring to a html page. 2.html
1 anchor tag referring to a html page. 1.html

You have to output this in a file:

1.html=>2

2.html=>1
3.html=>1

Name of the outputFile is pageFreq

All in all, you have to read 4 html files present in the folder HTMLPages
The names of the files are given above (page1.html , page2.html etc upto page4)

Let your output be sorted in lexicographic order alphabetically as above (as in the format followed in a dictionary).
Thanks to Madhudeep for discussing this idea with me :)

7: Question(S):

This Question is on the similar lines of the previous question.
If you've successfully done the previous question. This shouldn't come off as a big problem.

You must've observed that there were references from one page to another sequentially.
We want you to find out if there are any cyclic references in the html pages.

Ie: if page1 => page2 => page3 => page4 => page1

Write a html file saying cycle is present.
As in generate an html output file.

You can use this simple format

```
<!DOCTYPE html>
<html>
<body>

<h1>Output</h1>

<p>Cycle Present</p>

</body>
</html>
```

So your input files are present in 2 folders, read both the folders and

open the folder HTMLPages1
read everything in them and find out if a cyclic refernce is present in them

open the folder HTMLPages2
read everything in them and find out if a cyclic refernce is present in them

output shall be a html file.

in both cases you will be generating a html output file.
The name of the output file will be output1.html if the folder containing html pages have cycle references
The name of the output file will be output2.html if the folder containing html pages do not have cycle references.
Thanks to shweta for discussing the problem with me :)