**Tutorial and Practical 4 – Python Regular Expressions**

By the end of the lesson, you should be able to

* Explore and use Python regular expressions for pattern matching, searching and data extractions.

Task 1

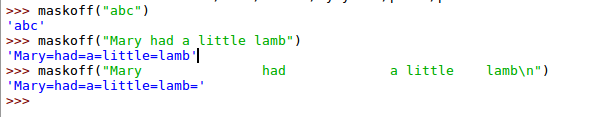
"Getting familiar with regular expression patterns"

1.1 Visit the interactive online page – " <http://regexone.com/>" and try to go through all the 14 basic lessons.

1.2a By using regular expressions, implement a function, maskoff() in python and it will :

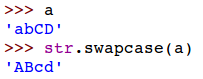
* Take in one string argument
* Return a new string which is based on the input argument but replace all the single or consecutive white spaces of the string with a "=" symbol from it.

Here is a sample call of maskoff() :



1.2b Python str provides an interesting function, swapcase(). It takes in a string and return the same string with all the alphabets swapped their cases.

e.g.



Now, implement a function, invertcases() in python and it will perform the same thing as swapcase:

* + You are not supposed to use swapcase.
  + You may try to explore if regex can help in this case.

Here is a sample call of invertcases():



Task 2

"Extract and Aggregate"

In this task, you will need to implement Python program to read in a long list of sales records from the file 'purchases.txt' and print out some aggregated information from the sales records.

The purchases.txt stores one sales record per line, and the lines look like this:

**2012-01-01 09:00 Austin Cameras 379.6 Visa**

There are total of 6 data fields, and they represent: date, time, store location, item category, sales value and payment method respectively, and they are separated by tabs.

2.1 Implement your program without using regular expressions and find out the following:

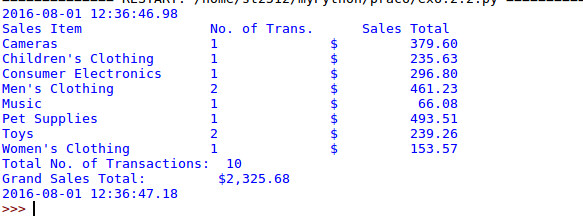
* The total number of sales entries.
* The total number of sales that using 'Amex' as the payment method.

2.2 Now try to implement the solution of 2.1 by using regular expressions. Do you find the regular expressions can help you to improve the performance of the program?

Reflection Prompt: State your views of these two approaches.

2.3 Implement a Python program to aggregate and print the aggregated number of transactions and total sales value that grouped by the sales item category.

Here is a sample run of the program for a small purchase.txt file that contains only 10 transaction records.



**~ The End ~**