Q1

Write a program (using functions!) that asks the user for a long string containing multiple words. Print back to the user the same string, except with the words in backwards order. For example, say I type the string:

My name is John

Then I would see the string:

John is name My

Challenge: One liner

Hints: Use join() and split() operations appropriately

Q2

FizzBuzz Off

Write a program that prints the numbers from 1 to 100. But for multiples of three print “Fizz” instead of the number and for the multiples of five print “Buzz”. For numbers which are multiples of both three and five print “FizzBuzz”.

Challenge: One liner (Logic)

Q3

What are the different elements?

Given two lists filled with positive integers, make a list of all elements which are not common among the two lists and print it.

Example:

Input:

2 3 4 5

1 2 3 4

Output:

5 1

Challenge: One liner (Logic)

Hint: Try using set theory ;) instead of looping

Q4

Best Fit Line

Write a program to accept a of input points(2D) and calculate the best fit line for it.

Hints

Consider a certain list of input points [(x_1, y_1) , (x_2, y_2) , (x_3, y_3) , ..., (x_n, y_n)]

Now, what is the best fit line ?

It is the line which tries to satisfy the given coordinates as best as it can. Given a certain x_i the value of $ax_i + b$ should be close to y_i where a and b are slope and y-intercept of the best fit line.

In other words, the sum total of the error : $((ax_i + b) - y_i)^2$ over all i should be minimized. This translates to a simple problem in multivariable calculus. You simply need to minimize the total error function with respect to the variables a and b. Work out the math and get a closed form for one of the variables in terms of data given and use that to calculate the optimal value of the other variable.

Make sure your code is as modular as possible. Break down the main statement using a top-down approach as discussed in the last lecture and proceed.....

Q5

Colliding Ball

In Software Development (especially Open Source development) one of the key skills of a good developer is to read and understand other people's code and make improvements to it. This could mean fixing issues that arises because of that code and/or new implementations thereby improving its functionality.

Let us start off simple. Look at the following code and try to get the gist of it (no , this has nothing to do with Github gist, I mean English gist here) : http://www.codeskulptor.org/#user43_5eXQYRtciE_5.py

Running the code tells us that collisions with the wall haven't been implemented yet. The ball just passes through. So, your task is to write the collision checks. All you need to do is to simply, write if-else conditional checks to see if the ball is within boundary limits. It should be only 4 - 5 lines of code.

Once you're done with this, go back to your rooms and experiment a bit more. How about more balls which can collide with each other ?

BONUS

Password Generator.

Be creative with how you generate passwords - strong passwords have a mix of lowercase letters, uppercase letters, numbers, and symbols. The passwords should be random, generating a new password every time the user asks for a new password.

Hint: Use random() and some predefined lists ;)