COMP-SCI 5590 – Python Lab 1

- **Author**

  Abdoh Jabbari

- **Objective**

  The objective of the lab is to be more familiar with some basic topics in python environments and to get to know how to solve some basic problems. Some of the basic topics in this lab are:

  - How to brain storm first and writing down the steps to understand the scenarios.
  - Find out what to use such as **List, String, Tuples, Functions, Loops, or if Conditions.**

- **Features**

  The main feature of this lab is basicness of the coding. In this lab we used different feature in each part of the lab. For example:

  Part-1: loop was used along with multiple if and elif conditions. Also, after each condition, break statements used to check each condition individually before moving into the second condition.

  Part-2: math library was used to help to count the number of words and find the middle words especially when the sentence is even number. If and else statement was used to apply conditions.

  Part-3: def was the feature used to declare the function name that is going to be used in this code. Also, loop with for statements along with if condition statements to compare the numbers to find three numbers equal to zero.

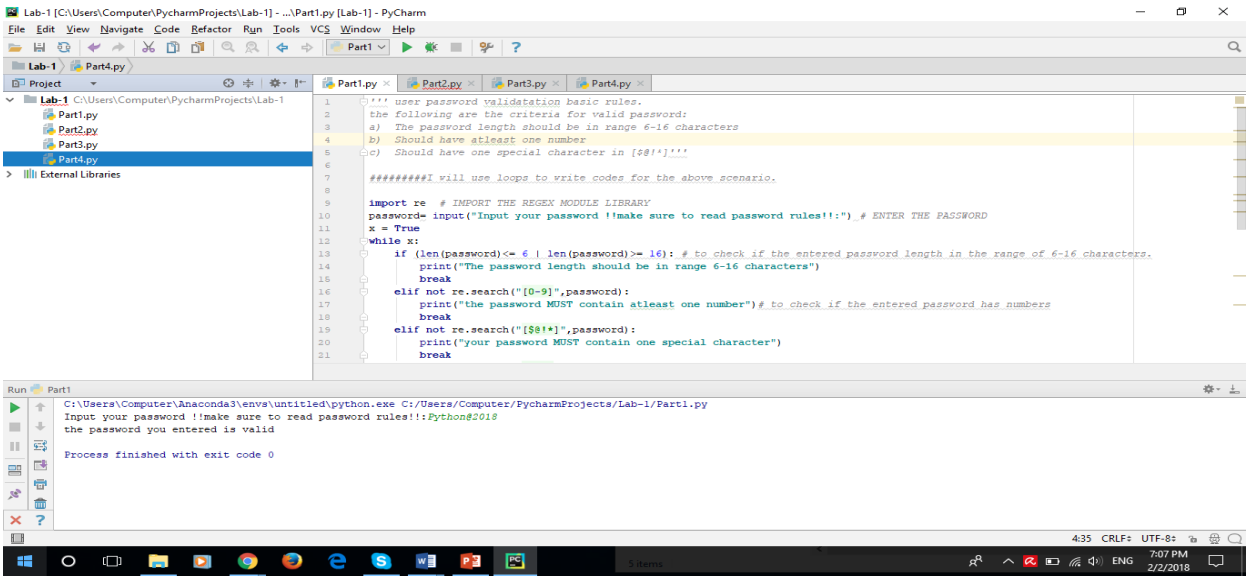
  Part-4: in this part the list of students was provided for both classes, to match between the two lists, loop feature was used along with if statements.

- **Configuration**

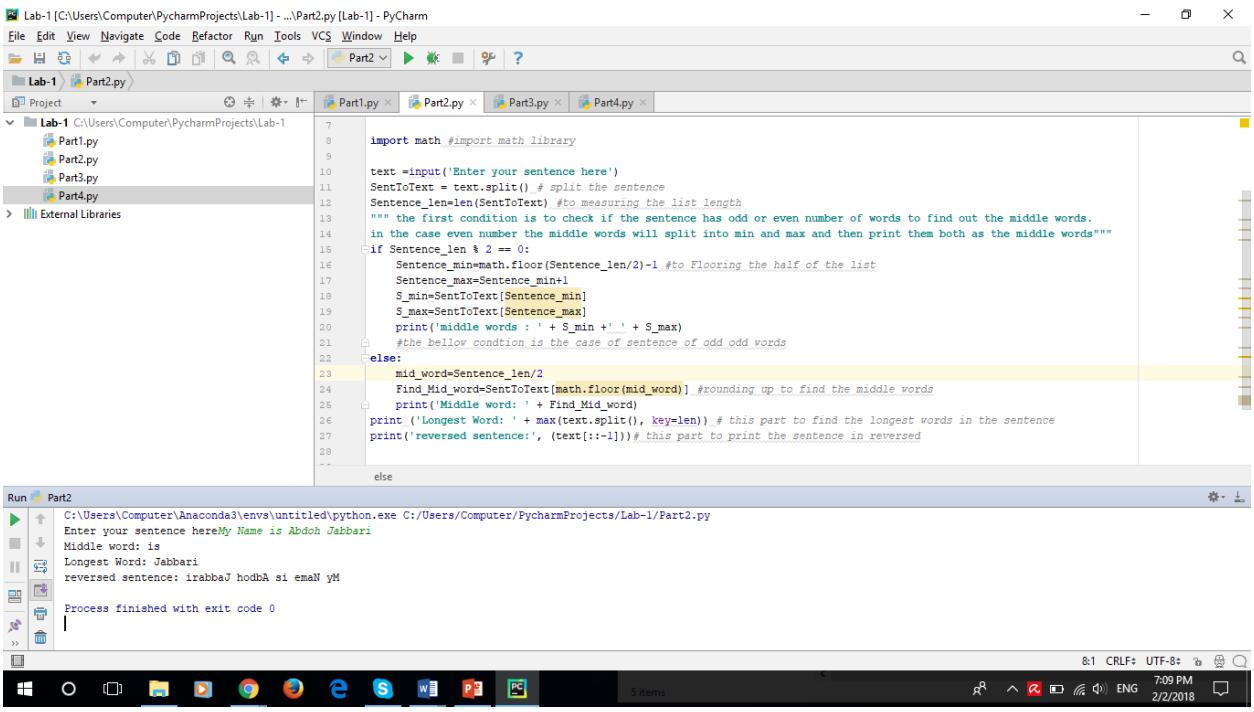
The configuration environment is python 3 as the interpreter and the IDE used in this lab is pycharm.

- **Input/output (screenshots)**

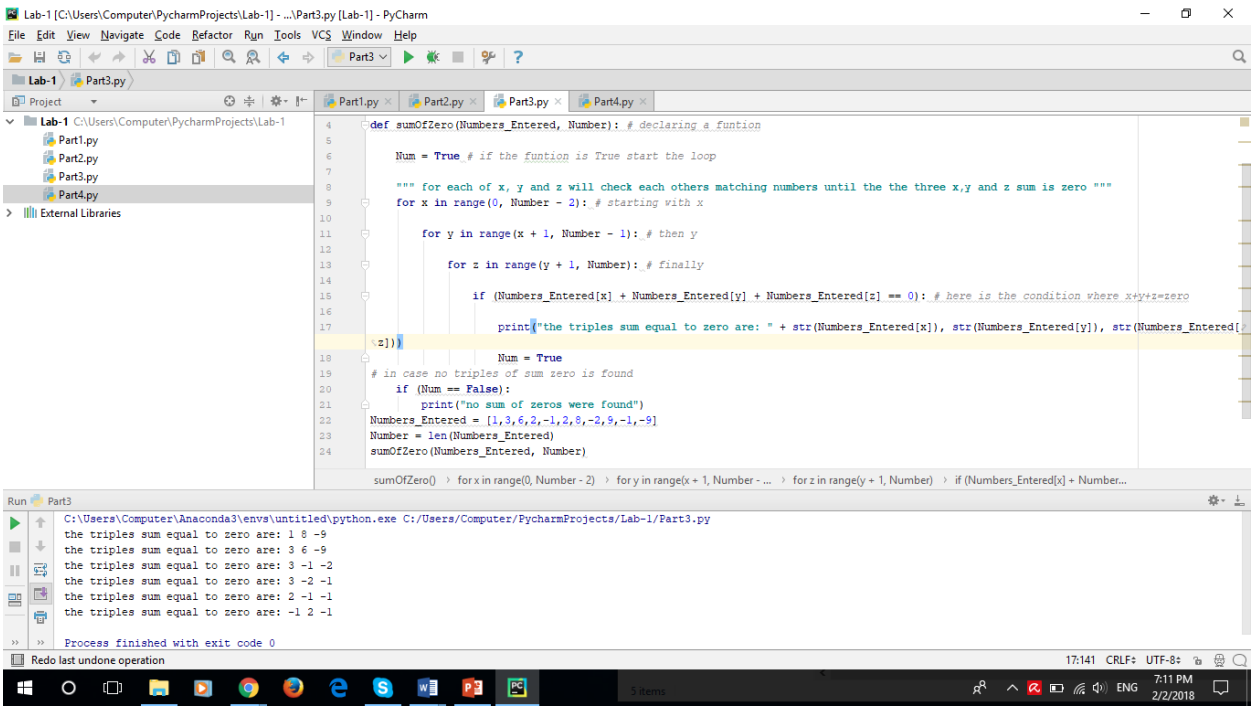
Part-1



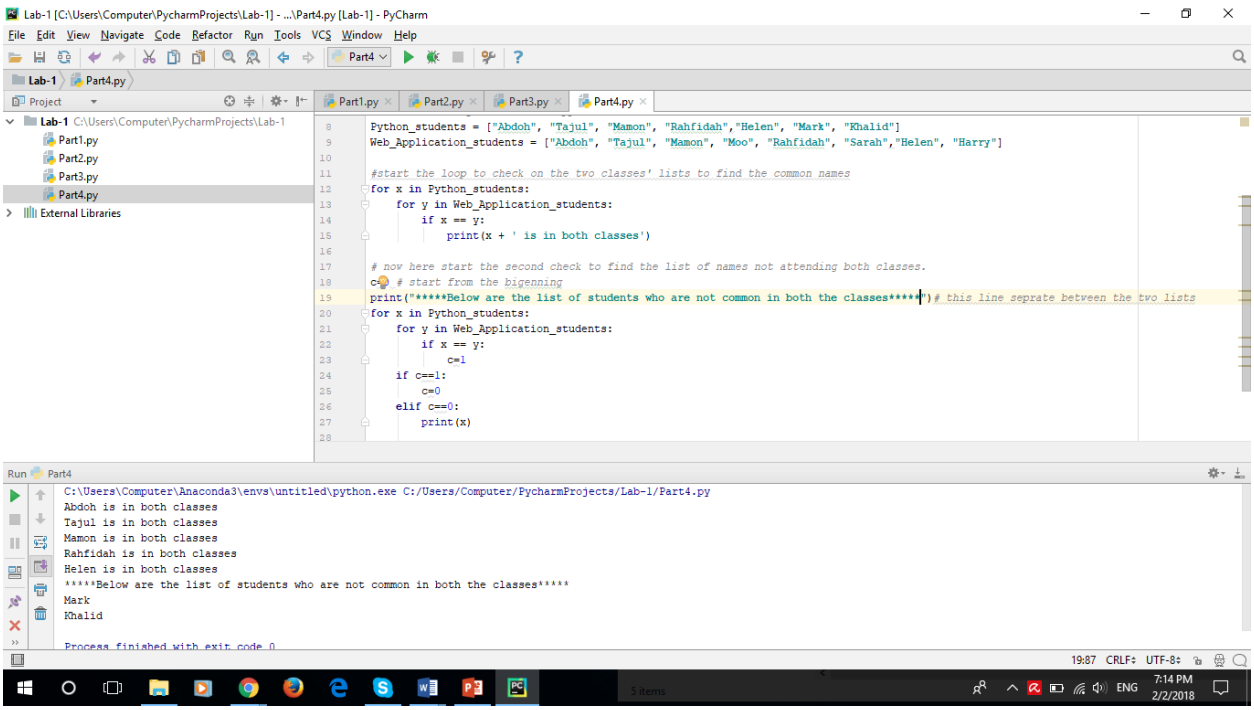
Part-2



Part-3


Lab-1 [C:\Users\Computer\PycharmProjects\Lab-1] - ...\Part3.py [Lab-1] - PyCharm
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Part1.py Part2.py Part3.py Part4.py
def sumOfZero(Numbers_Entered, Number): # declaring a funtion
Num = True # if the funtion is True start the loop
""" for each of x, y and z will check each others matching numbers until the the three x,y and z sum is zero """
for x in range(0, Number - 2): # starting with x
for y in range(x + 1, Number - 1): # then y
for z in range(y + 1, Number): # finally
if (Numbers_Entered[x] + Numbers_Entered[y] + Numbers_Entered[z] == 0): # here is the condition where x+y+z=zero
print("the triples sum equal to zero are: " + str(Numbers_Entered[x]), str(Numbers_Entered[y]), str(Numbers_Entered[z]))
Num = True
# in case no triples of sum zero is found
if (Num == False):
print("no sum of zeros were found")
Numbers_Entered = [1,3,6,2,-1,2,8,-2,9,-1,-9]
Number = len(Numbers_Entered)
sumOfZero(Numbers_Entered, Number)
Run Part3
C:\Users\Computer\Anaconda3\envs\untitled\python.exe C:/Users/Computer/PycharmProjects/Lab-1/Part3.py
the triples sum equal to zero are: 1 8 -9
the triples sum equal to zero are: 3 6 -9
the triples sum equal to zero are: 3 -1 -2
the triples sum equal to zero are: 3 -2 -1
the triples sum equal to zero are: 2 -1 -1
the triples sum equal to zero are: -1 2 -1
Process finished with exit code 0

Part-4


Lab-1 [C:\Users\Computer\PycharmProjects\Lab-1] - ...\Part4.py [Lab-1] - PyCharm
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Part1.py Part2.py Part3.py Part4.py
Python_students = ["Abdoh", "Tajul", "Mamon", "Rahfidah","Helen", "Mark", "Khalid"]
Web_Application_students = ["Abdoh", "Tajul", "Mamon", "Moo", "Rahfidah", "Sarah","Helen", "Harry"]
#start the loop to check on the two classes' lists to find the common names
for x in Python_students:
for y in Web_Application_students:
if x == y:
print(x + ' is in both classes')
# now here start the second check to find the list of names not attending both classes.
c=0 # start from the bigenning
print("*****Below are the list of students who are not common in both the classes*****")# this line seprate between the two lists
for x in Python_students:
for y in Web_Application_students:
if x == y:
c=1
if c==1:
c=0
elif c==0:
print(x)
Run Part4
C:\Users\Computer\Anaconda3\envs\untitled\python.exe C:/Users/Computer/PycharmProjects/Lab-1/Part4.py
Abdoh is in both classes
Tajul is in both classes
Mamon is in both classes
Rahfidah is in both classes
Helen is in both classes
*****Below are the list of students who are not common in both the classes*****
Mark
Khalid
Process finished with exit code 0

- **<u>Explain the implementation including code snippet</u>**

  For each part I did use different implementation; for example:

  Part-1 I use loop to check the entered phrase if it is meeting the list of the rules:

  a) *check if the length in range 6-16 characters*
  b) *check if it has at least one number*
  c) *check if it has one special character in [$@!*]'''*
  d) *check if it has at least one lowercase[a-z]*
  e) *check if it has at least one uppercase character[A-Z]*

```
import re  # IMPORT THE REGEX MODULE LIBRARY
password= input("Input your password !!make sure to read password rules!!:") # ENTER THE PASSWORD
x = True
while x:
    if (len(password)<= 6 | len(password)>= 16): # to check if the entered password length in the range of 6-16 characters.
        print("The password length should be in range 6-16 characters")
        break
    elif not re.search("[0-9]",password):
        print("the password MUST contain atleast one number")# to check if the entered password has numbers
        break
    elif not re.search("[$@!*]",password):
        print("your password MUST contain one special character")
        break
    elif not re.search("[a-z]", password):
        print(
            "your password must contain a lowercase letter")  # to check if the entered password has lowercase character
        break
    elif not re.search("[A-Z]", password):
        print(
            "your password must contain an upercase letter")  # to check if the entered password has uppercase character
        break
    else: # if the password passed all the above, issue the password
        print("the password you entered is valid")

        # if the passowrd is not meeting any of the requirements tell the user to try again then go out of the loop.

        x=False
        break
if x:
    print("please try to recreate your password again!!!")
```

Part-2

In this part I needed to use math library to help me figure out the middle words because I had problem solving it at the beginning especially when the words in the sentence is even. I split the words in the sentence then count the words then to find the middle words I divide the middle into min and max value in the case of even number then print them both to be the middle words. In the case of odd it has no problems to find the middle word. The next

two section of this part I didn't need to use if condition as the first one. I just len() function to find the longest word and I used [::-1] to print the sentence in reverse order.

```
import math #import math library

text =input('Enter your sentence here')
SentToText = text.split() # split the sentence
Sentence_len=len(SentToText) #to measuring the list length
""" the first condition is to check if the sentence has odd or even number of words to find out the middle words.
in the case even number the middle words will split into min and max and then print them both as the middle words"""
if Sentence_len % 2 == 0:
    Sentence_min=math.floor(Sentence_len/2)-1 #to Flooring the half of the list
    Sentence_max=Sentence_min+1
    S_min=SentToText[Sentence_min]
    S_max=SentToText[Sentence_max]
    print('middle words : ' + S_min +' ' + S_max)
    #the bellow condtion is the case of sentence of odd odd words
else:
    mid_word=Sentence_len/2
    Find_Mid_word=SentToText[math.floor(mid_word)] #rounding up to find the middle words
    print('Middle word: ' + Find_Mid_word)
print ('Longest Word: ' + max(text.split(), key=len)) # this part to find the longest words in the sentence
print('reversed sentence:', (text[::-1]))# this part to print the sentence in reversed
```

Part 3.

This part of the lab is to find triplets in the list which gives the sum of zero, to do such I needed to define three variables x, y and z, then for each variable it will be representing the list of numbers. Then each variable each will check on other values until we meet the condition x+y+z=0. If the condition did not meet it will move to the other condition which is print the sum of zero not found.

```
#To to find triplets in the list which gives the sum of zero.

def sumOfZero(Numbers_Entered, Number): # declaring a funtion

    Num = True # if the funtion is True start the loop

    """ for each of x, y and z will check each others matching numbers until the the three x,y and z sum is zero """
    for x in range(0, Number - 2): # starting with x

        for y in range(x + 1, Number - 1): # then y

            for z in range(y + 1, Number): # finally

                if (Numbers_Entered[x] + Numbers_Entered[y] + Numbers_Entered[z] == 0): # here is the condition where x+y+z=zero

                    print("the triples sum equal to zero are: " + str(Numbers_Entered[x]), str(Numbers_Entered[y]), str(Numbers_Entered[z]))
                    Num = True
# in case no triples of sum zero is found
    if (Num == False):
        print("no sum of zeros were found")
Numbers_Entered = [1,3,6,2,-1,2,8,-2,9,-1,-9]
Number = len(Numbers_Entered)
sumOfZero(Numbers_Entered, Number)
```

Part 4

In this part we have list of students who are attending class “Python” and another list of students who are attending class “Web Application” and we need to find who adding both classes and who is not. I used loop to pick each name from python and compare it to the list on web application students’ list. If the match found in both then it prints it out, and then go back again do the same process gain with the second name until the list is done. Then for find the none common name we use the condition in which if the name is not found to be matching from python class with web application.

```
""" Consider the following scenario. You have a list of students who are attending
class "Python" and another list of students who are attending class "Web Application".
****find the list of students who are attending both the classes.
Also find the list of students who are not common in both the classes."""

# the list of Python and Web application classes students
Python_students = ["Abdoh", "Tajul", "Mamon", "Rahfidah","Helen", "Mark", "Khalid"]
Web_Application_students = ["Abdoh", "Tajul", "Mamon", "Moo", "Rahfidah", "Sarah","Helen", "Harry"]

#start the loop to check on the two classes' lists to find the common names
for x in Python_students:
    for y in Web_Application_students:
        if x == y:
            print(x + ' is in both classes')

# now here start the second check to find the list of names not attending both classes.
c=0 # start from the bigenning
print("*****Below are the list of students who are not common in both the classes*****")# this line seprate between the two lists
for x in Python_students:
    for y in Web_Application_students:
        if x == y:
            c=1
    if c==1:
        c=0
    elif c==0:
        print(x)
```

- **<u>Explain about the deployment</u>**

  The deployment in each part was done in couple of steps:

  - Reading the scenarios or the problems and understand it
  - Write down the main ideas and functions needed to be used
  - Start writing the code
  - Execute the code to check the results
  - Write the comments on the code.

- **<u>Limitation</u>**

The limitation in this lab could be the basicness of writing my code and for sure there are a lot better codes than I have in this lab.

- **References**

I have used couple of resources such github, stackoverflow, teamtreehouse, and python official webpage, to get this lab done. I also watched a couple of videos to manage to get through with this lab. Below is the list of those references:

a) http://www.programmingforums.org/thread27655.html
b) https://github.com/fnl/segtok
c) http://www.pythonforbeginners.com/basics/string-manipulation-in-python
d) https://www.geeksforgeeks.org/find-triplets-array-whose-sum-equal-zer
e) https://teamtreehouse.com/community/longest-word-in-a-list
f) https://www.tutorialspoint.com/python/number_floor.htm
g) https://stackoverflow.com/questions/19410018/how-to-count-the-number-of-words-in-a-sentence
h) https://stackoverflow.com/questions/29686952/nested-loops-to-determine-students-in-two-classes-bot
i) https://stackoverflow.com/questions/19410018/how-to-count-the-number-of-words-in-a-sentence