PYTHON LAB ASSIGNMENT – 1

Sai Mohith Reddy Chagamreddy

Class ID - 9

Objective: The main objective of this lab assignment is to get familiar with the basic topics in python like functions, lists, tuples, loops etc.

Features:

The programs in the offers the following features:

1. validate password

2. get the middle, longest and sentence with reversed words

3. get the triplets which gives sum zero from the given input sequence

4. get the students common and not common from the course lists

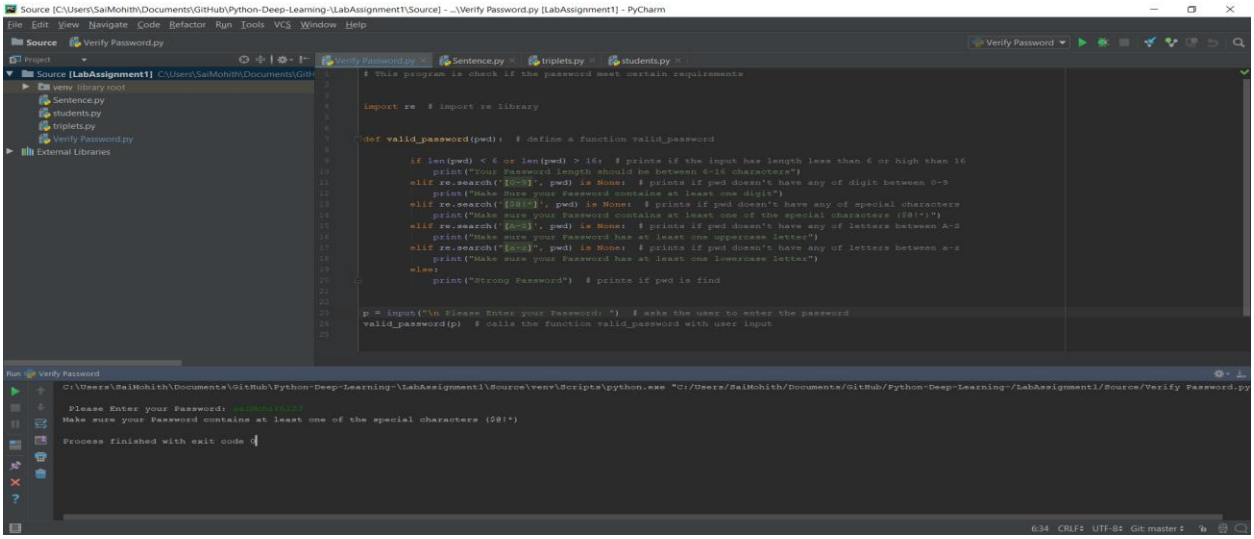
Configuration:

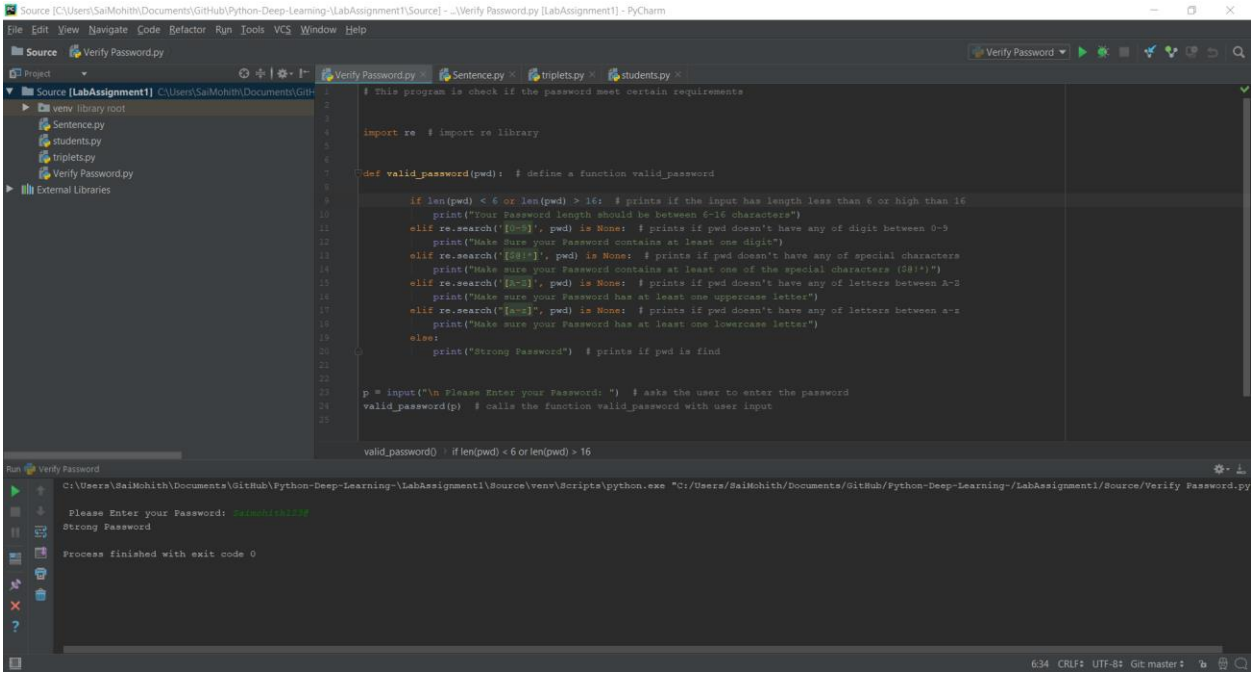
Python 3.6 interpreter

JetBrains Pycharm Community Edition

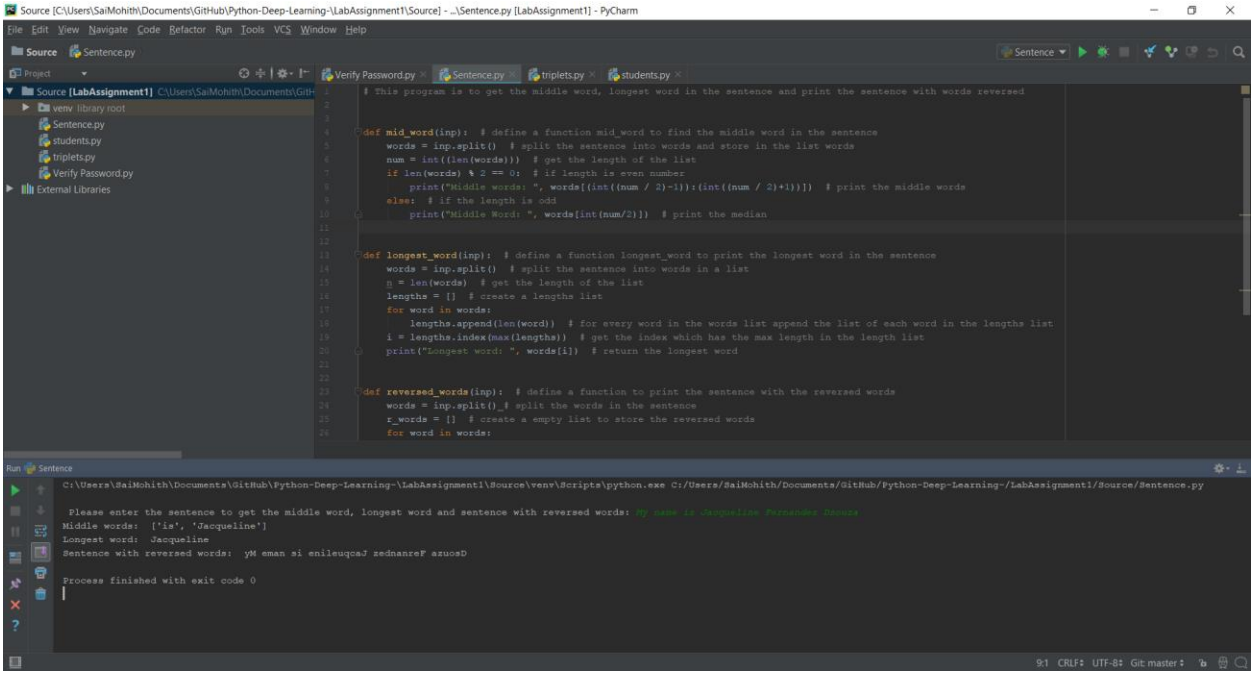
Screenshots:

1. Valid Password Input and Output:

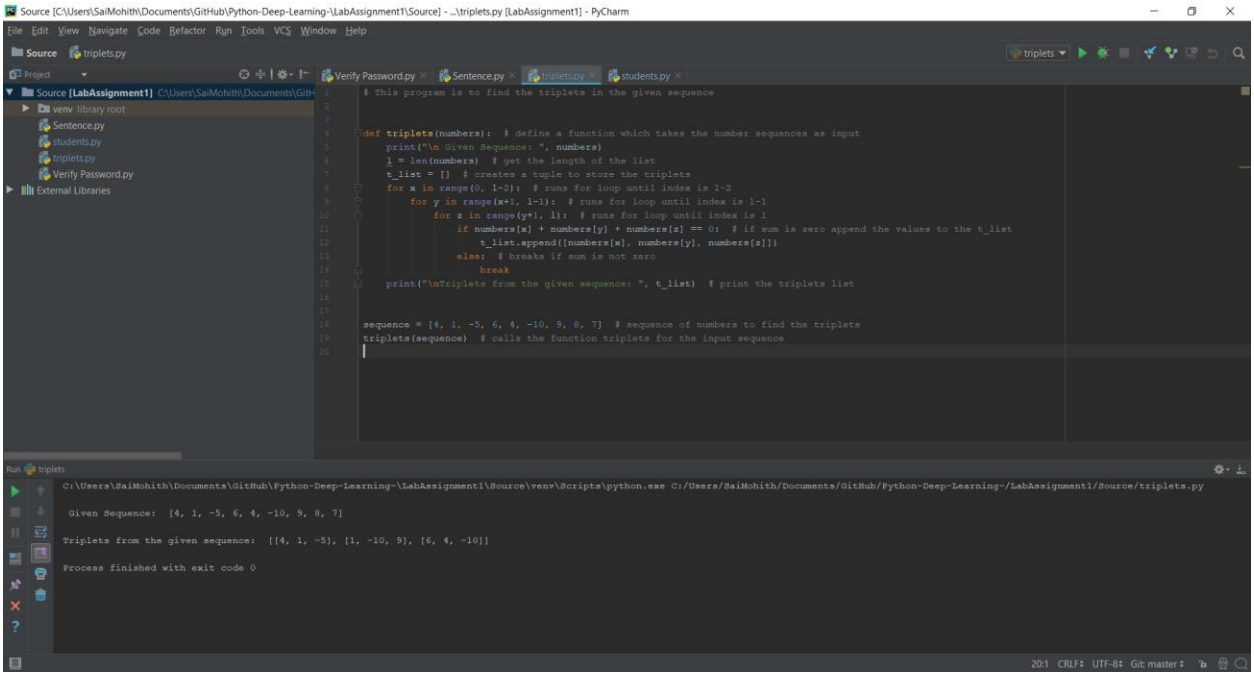




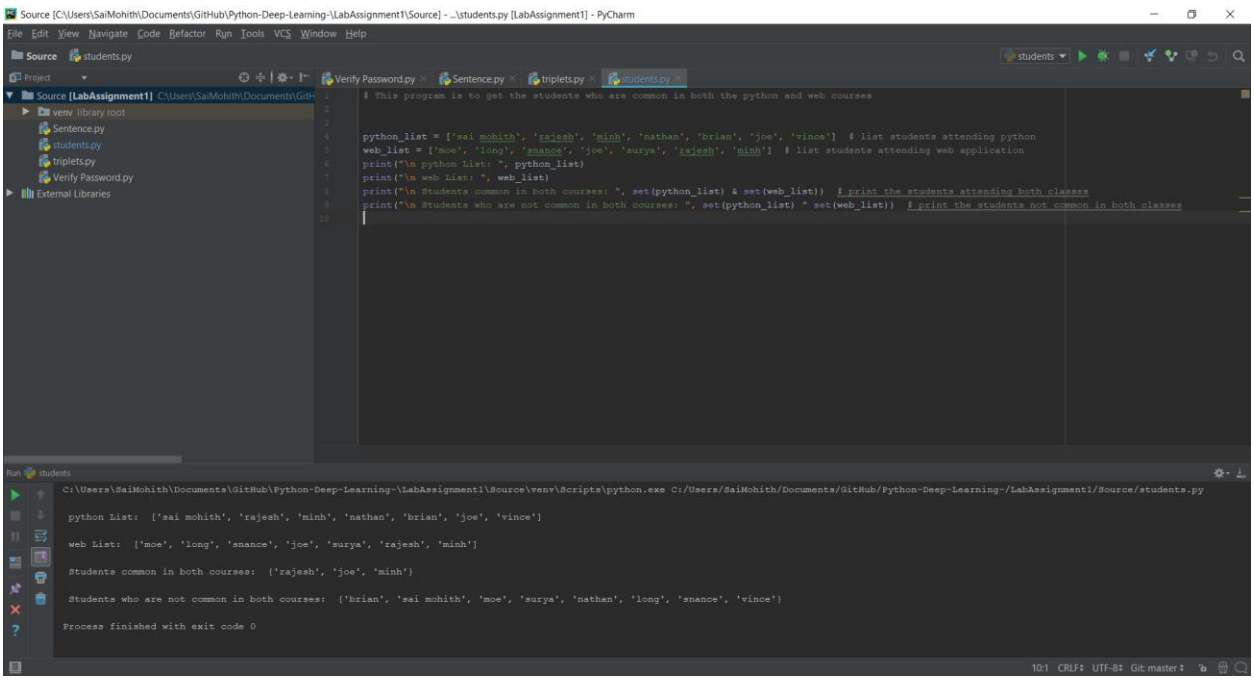
2. Middle, longest words and sentence with reversed words:



3. Triplets in the given sequence of numbers:



4. Students list output:



Code Implementation:

1. Valid Password program:

This program is to verify whether the password given by the user meets the requirements or not. It displays the missing requirements in the password to the user as the input is given. If the password meets all the rules, it displays **Strong Password**

We first import the library re to use search function for searching the required characters in the input.

```
import re  # import re library
```

We define a function here to check whether the password has met all the rules

```
def valid_password(pwd):  # define a function valid_password

    if len(pwd) < 6 or len(pwd) > 16:  # prints if the input has length less than 6 or high than 16
        print("Your Password length should be between 6-16 characters")
    elif re.search('[0-9]', pwd) is None:  # prints if pwd doesn't have any of digit between 0-9
        print("Make Sure your Password contains at least one digit")
    elif re.search('[$@!*]', pwd) is None:  # prints if pwd doesn't have any of special characters
        print("Make sure your Password contains at least one of the special characters ($@!*)")
    elif re.search('[A-Z]', pwd) is None:  # prints if pwd doesn't have any of letters between A-Z
        print("Make sure your Password has at least one uppercase letter")
    elif re.search("[a-z]", pwd) is None:  # prints if pwd doesn't have any of letters between a-z
        print("Make sure your Password has at least one lowercase letter")
    else:
        print("Strong Password")  # prints if pwd is find
```

Asks the user to give input

```
p = input("\n Please Enter your Password: ")  # asks the user to enter the password
```

After user enters the input, the function will be called

```
valid_password(p)  # calls the function valid_password with user input
```

2. Middle, longest words and sentence with reversed words:

To get the middle word from the given sentence we first define a function and check the middle word as shown in the below code.

```
def mid_word(inp):  # define a function mid_word to find the middle word in the sentence
    words = inp.split()  # split the sentence into words and store in the list words
    num = int((len(words)))  # get the length of the list
    if len(words) % 2 == 0:  # if length is even number
        print("Middle words: ", words[(int((num / 2)-1)):(int((num / 2)+1))])  # print the middle words
    else:  # if the length is odd
        print("Middle Word: ", words[int(num/2)])  # print the median
```

To get the longest word in the given sentence we define a function and get the longest sentence by checking the length of each word and append to a list. Then print the word with highest length using max function.

```
def longest_word(inp):  # define a function longest_word to print the longest word in the sentence
    words = inp.split()  # split the sentence into words in a list
```

```
    lengths = []  # create a lengths list
    for word in words:
        lengths.append(len(word))  # for every word in the words list append the list of each word in the lengths list
    i = lengths.index(max(lengths))  # get the index which has the max length in the length list
    print("Longest word: ", words[i])  # return the longest word
```

To get the reversed words, we first split the sentence and reverse the words. Then, we join the list with reversed words with join function.

```
def reversed_words(inp):  # define a function to print the sentence with the reversed words
    words = inp.split() # split the words in the sentence
    r_words = []  # create a empty list to store the reversed words
    for word in words:
        r_words.append(word[::-1])  # for each word in words get the reversed word and append it in the r_words list
    print("Sentence with reversed words: ", ' '.join(r_words))  # print the joined version of the r_words list
```

Then we ask the user for input and call the functions with the input from the user.

```
sent = input("\n Please enter the sentence to get the middle word, longest word and sentence with reversed words:")
mid_word(sent)  # call the mid_word function for the user input
longest_word(sent)  # call the longest_word function
reversed_words(sent)  # call the reversed_words function
```

3. Triplets:

This program is to find the triplets which gives a sum of zero from the given sequence of numbers by the user. We first define a function to find triplets. In the function we write the code to find the length of the array. Using for loops we calculate the sum of all the triplets in the given array. We append the triplets which gives a sum of zero to a new empty list and print that list. This is shown in the below code:

```
def triplets(numbers):  # define a function which takes the number sequences as input
    l = len(numbers)  # get the length of the list
    t_list = []  # creates a tuple to store the triplets
    for x in range(0, l-2):  # runs for loop until index is l-2
        for y in range(x+1, l-1):  # runs for loop until index is l-1
            for z in range(y+1, l):  # runs for loop until index is l
                if numbers[x] + numbers[y] + numbers[z] == 0:  # if sum is zero append the values to the t_list
                    t_list.append([numbers[x], numbers[y], numbers[z]])
                else:  # breaks if sum is not zero
                    break
    print("\nTriplets from the given sequence: ", t_list)  # print the triplets list
```

Now we call the function with the users input

```
sequence = [4, 1, -5, 6, 4, -10, 9, 8, 7]  # sequence of numbers to find the triplets
triplets(sequence)  # calls the function triplets for the input sequence
```

4. Student List:

This program gives the students attending both python and web courses and those who are not in common with both courses.

For this we first take the list of students attending python and a list of students attending web course. Now a common list of students can be obtained using "&" operator. we use "^" to find the students not common in both courses.

```
python_list = ['sai mohith', 'rajesh', 'minh', 'nathan', 'brian', 'joe', 'vince']  # list students attending python

web_list = ['moe', 'long', 'snance', 'joe', 'surya', 'rajesh', 'minh']  # list students attending web application

print("Students common in both courses: ", set(python_list) & set(web_list))  # print the students attending both

classes

print("Students who are not common in both courses: ", set(python_list) ^ set(web_list))  # print the students not

common in both classes
```

Code Deployment:

1. Valid Password program:

Program output with input as saiMohith123 will print the output "Make sure your Password contains at least one of the special characters ($@!*)" as there is no special character in the input. When the input is given Saimohith123@, the output will be "Strong Password" as it has met all the requirements.

```
Please Enter your Password: saiMohith123

Make sure your Password contains at least one of the special characters ($@!*)

Please Enter your Password: Saimohith123@

Strong Password
```

2. Middle, longest words and sentence with reversed words:

The output for the program with the input as "My Name is Jacqueline Fernandez Dsouza" is seen below

Please enter the sentence to get the middle word, longest word and sentence with reversed words: My Name is Jacqueline Fernandez Dsouza

Middle words: ['is', 'Jacqueline']

Longest word: Jacqueline

Sentence with reversed words: yM emaN si enileuqcaJ zednanreF azuosD

Process finished with exit code 0

3. Triplets:

The triplets which gives sum of zero for the given input sequence [4, 1, -5, 6, 4, -10, 9, 8, 7] is shown below:

Triplets from the given sequence: [[4, 1, -5], [1, -10, 9], [6, 4, -10]]

Process finished with exit code 0

4. Student List:

Input lists are:

python_list = ['sai mohith', 'rajesh', 'minh', 'nathan', 'brian', 'joe', 'vince'] # list students attending python

web_list = ['moe', 'long', 'snance', 'joe', 'surya', 'rajesh', 'minh'] # list students attending web application

The output of the student list for given lists of two courses will be:

Students common in both courses: {'minh', 'joe', 'rajesh'}

Students who are not common in both courses: {'long', 'nathan', 'brian', 'sai mohith', 'snance', 'vince', 'surya',

'moe'}

Process finished with exit code 0

Limitations:

This code is written in python version 3.6. Some of the functions used in these codes may not work when deployed in the python 2.x.

References:

https://www.pythonlearn.com/html-008/cfbook005.html

https://docs.python.org/3.3/tutorial/datastructures.html

https://docs.python.org/3.3/tutorial/controlflow.html

www.stackoverflow.com