**Python Lab Assignment -1**

**Name – Aravind Sheri, Class ID – 31**

**Name – Rajiv Varma, Class ID – 04**

**Team ID - 19**

**1.**

**Introduction:**

Validating the rules for the entered password by the user using various scenarios using loops.

**Objectives:**

Getting the password entered by the user and validate against the rules and update the user to get the correct password according to the rules and satisfy all the rules.

**Approach:**

Created regular expressions for the number, special character, and Upper and lower case where at least one character in the password should contain all of the above characters and user if and elseif loops to check each and every condition.

**Workflow:**

User will enter the password and the password is validated against all the use cases which are defined in the various if and elseif statements and if the password meets all the conditions the user will get a success message.

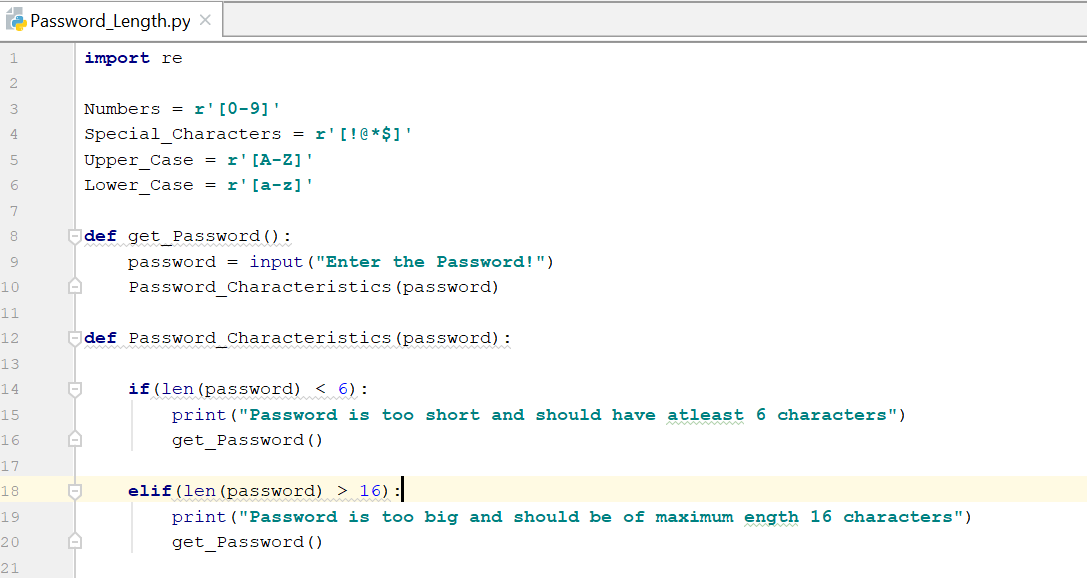
**Parameters:**

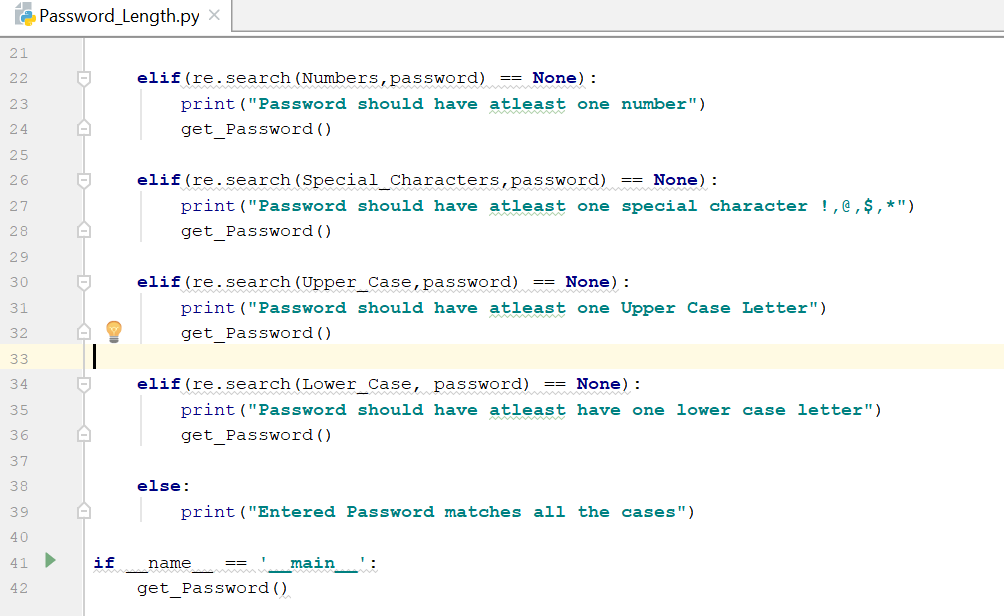
Parameters is only the password and it is entered by User.

**Conclusion:**

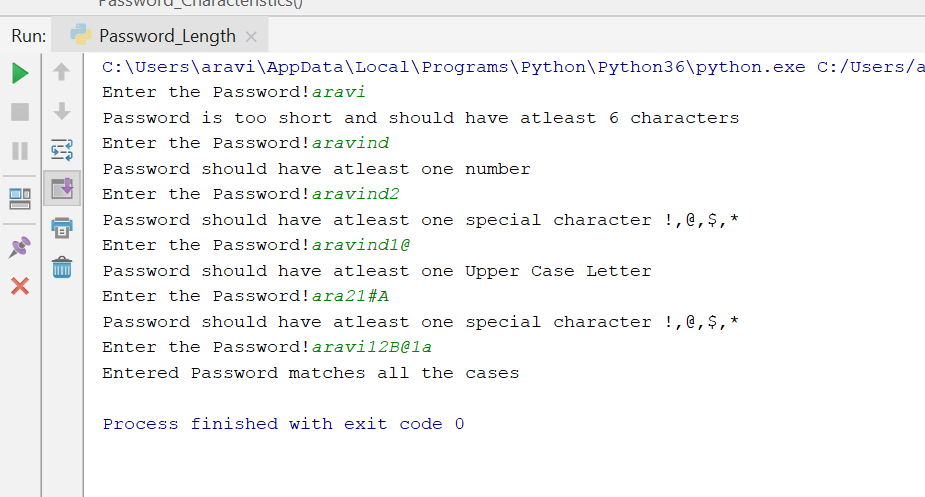
The code worked fie and it will notify if the entered password is not satisfying the scenarios.

**Code:**





**Output:**



**2.**

**Introduction:**

Accepting a sentence and print the longest word, middle word and the sentence with reversed words

**Objectives:**

Modifying the entered string according to the above rules and print the output

**Approach:**

Split the sentence on space and getting the individual words and using the string length which is either even or odd. If the length is even two middle words will be printed and if odd middle word is printed.

**Workflow:**

User will be entering the sentence and various actions will be performed and the output is printed on the console.

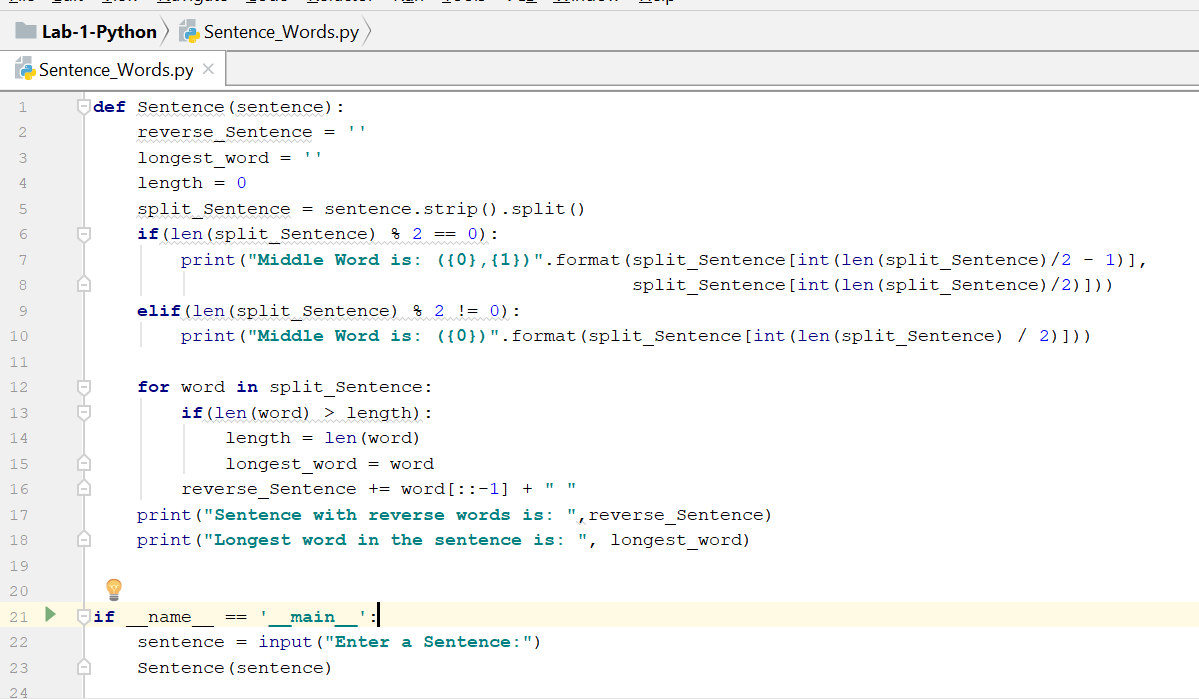
**Parameters:**

Parameters is only sentence entered by the user.

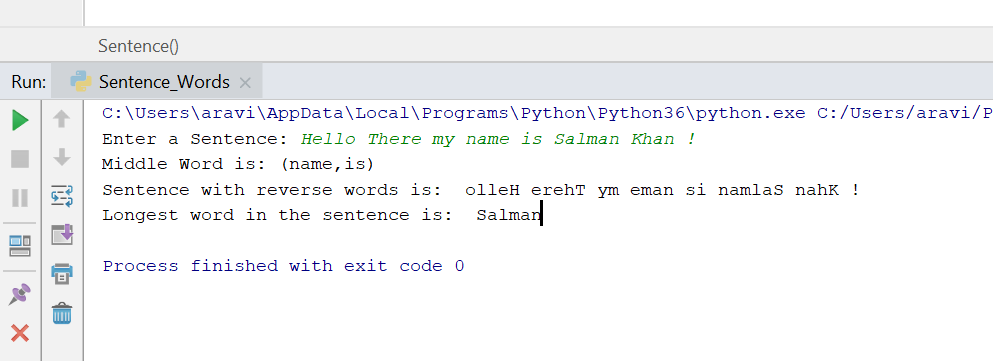
**Conclusion:**

The code worked fine and it will print the sentence according to the above scenarios.

Code:



**Output:**



**3.**

**Introduction:**

Finding the triplets in a list whose sum is Zero.

**Objectives:**

Iterate through the list and find the triplets whose sum is zero.

**Approach:**

Sorted the array and took the first two elements and iterated through the rest of list and added the sum of first two elements with the first element in the rest of the list.

**Workflow:**

Input is given as list and the we find the length, if the length is less than 3 the code will exit. Else it will iterate through the list and find the triplets whose sum is zero.

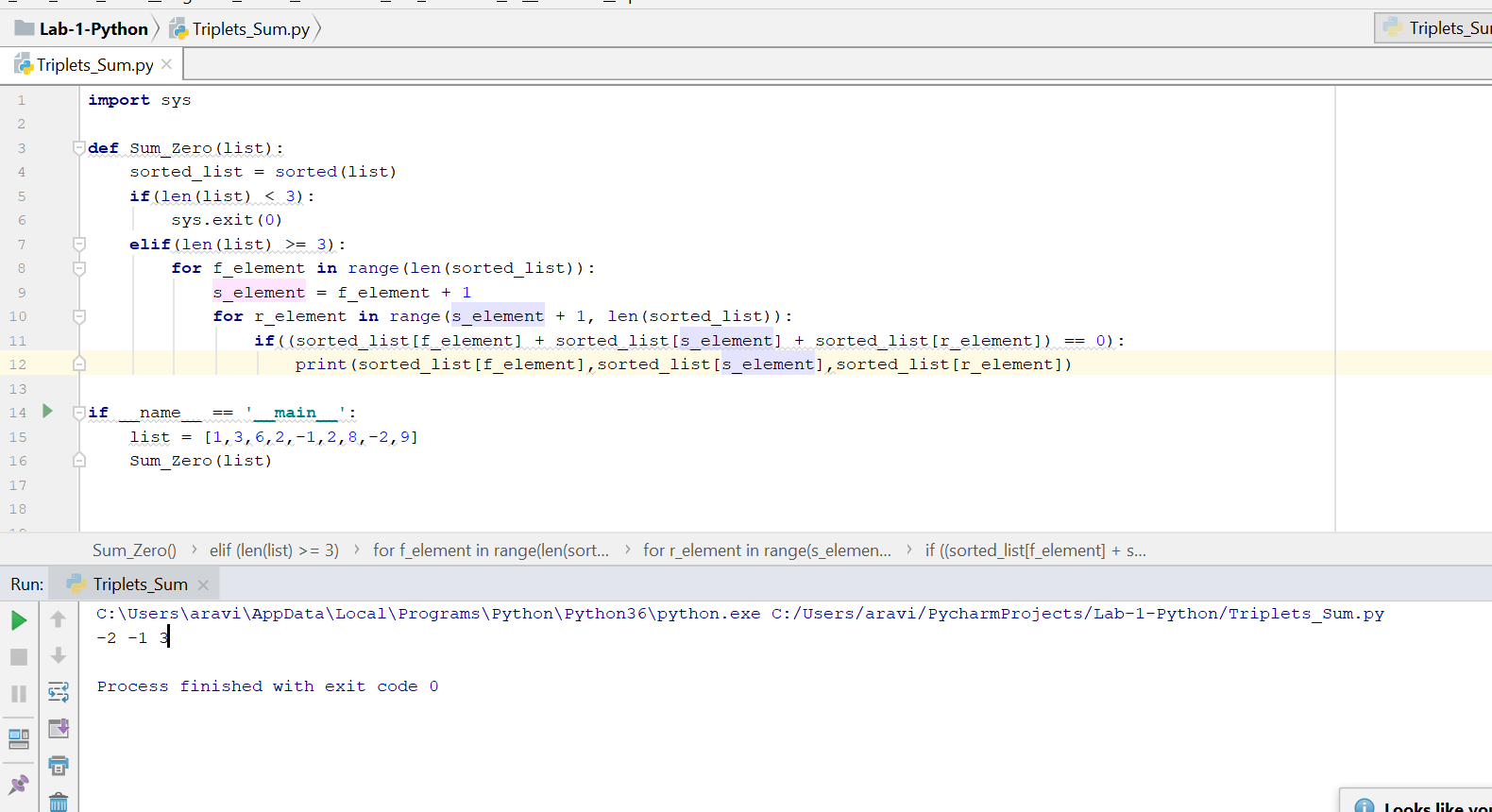
**Parameters:**

No Parameters

**Conclusion:**

The code worked fine, and it will print the 3 elements of the list whose sum is zero.

Code and output:



**4.**

**Introduction:**

Finding the students who are common and uncommon in the classes Python and Big Data which they have enrolled.

**Objectives:**

Iterate through the list Python students and Big Data students and find the common and un common students.

**Approach:**

Compare each element in list Python with the elements in Big Data and if the element is present add it to common list and delete it from Big Data list. Else if element is not present add it to the un common list. Concatenate the un common list with the remaining un common list in Big Data.

**Workflow:**

Two lists are given, and it will take each element from Python and compare with elements in Big Data list and produces common and un common lists.

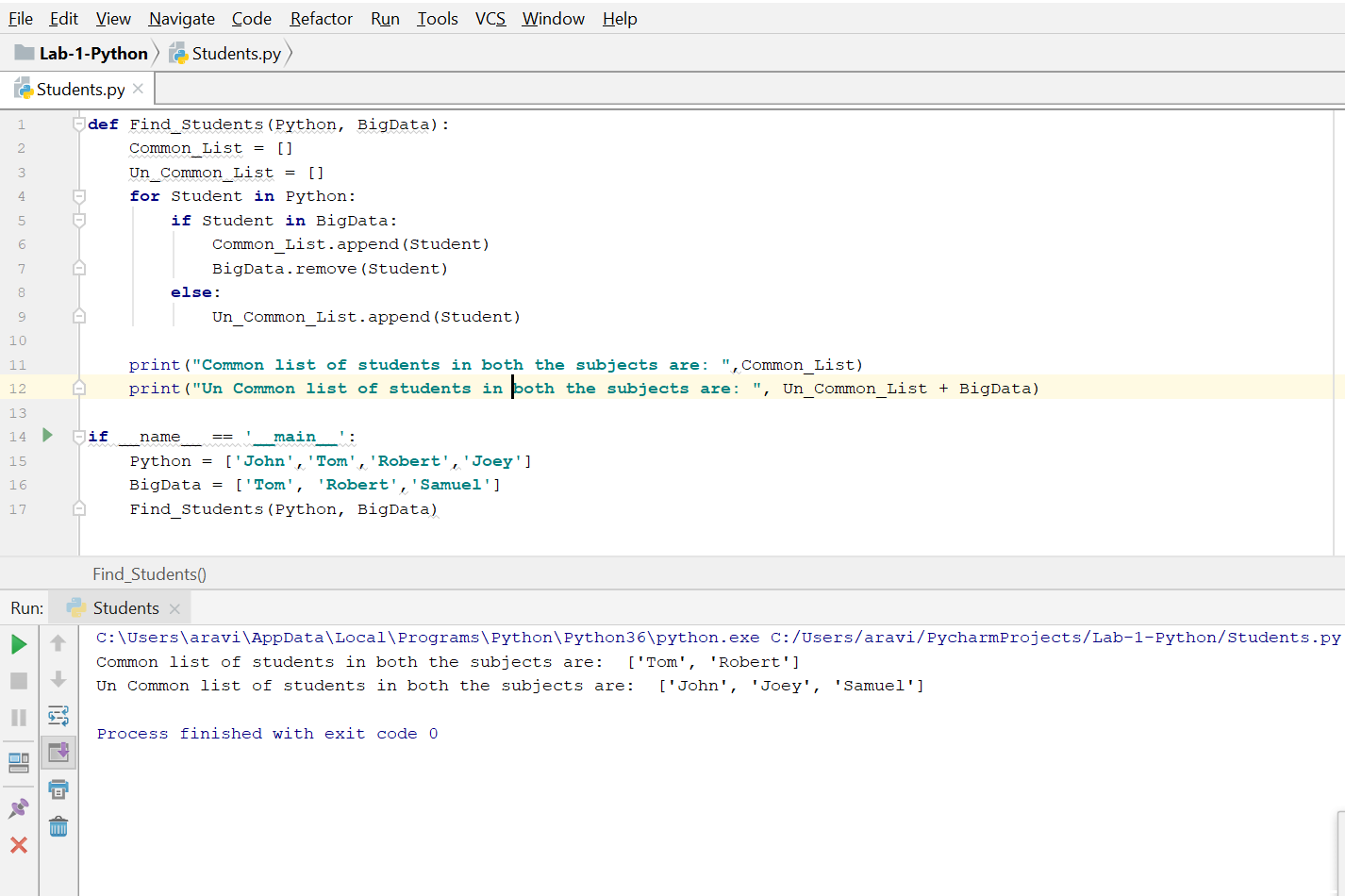
**Parameters:**

Lists of students enrolled in two subjects

**Conclusion:**

The code worked fine, and it will print the common and uncommon students.

**Code and Output:**



**5.**

**Introduction:**

Air way Booking system and implementing the class concepts and various inheritance concepts.

**Objectives:**

Creating classes of various levels of airport booking system and implementing the relationships between them.

**Approach:**

Created 5 classes and implemented various relationships between them. Created objects for every member in the class and printed various results.

**Workflow:**

**Created various objects and called the various functionalities in the program and printed the output**

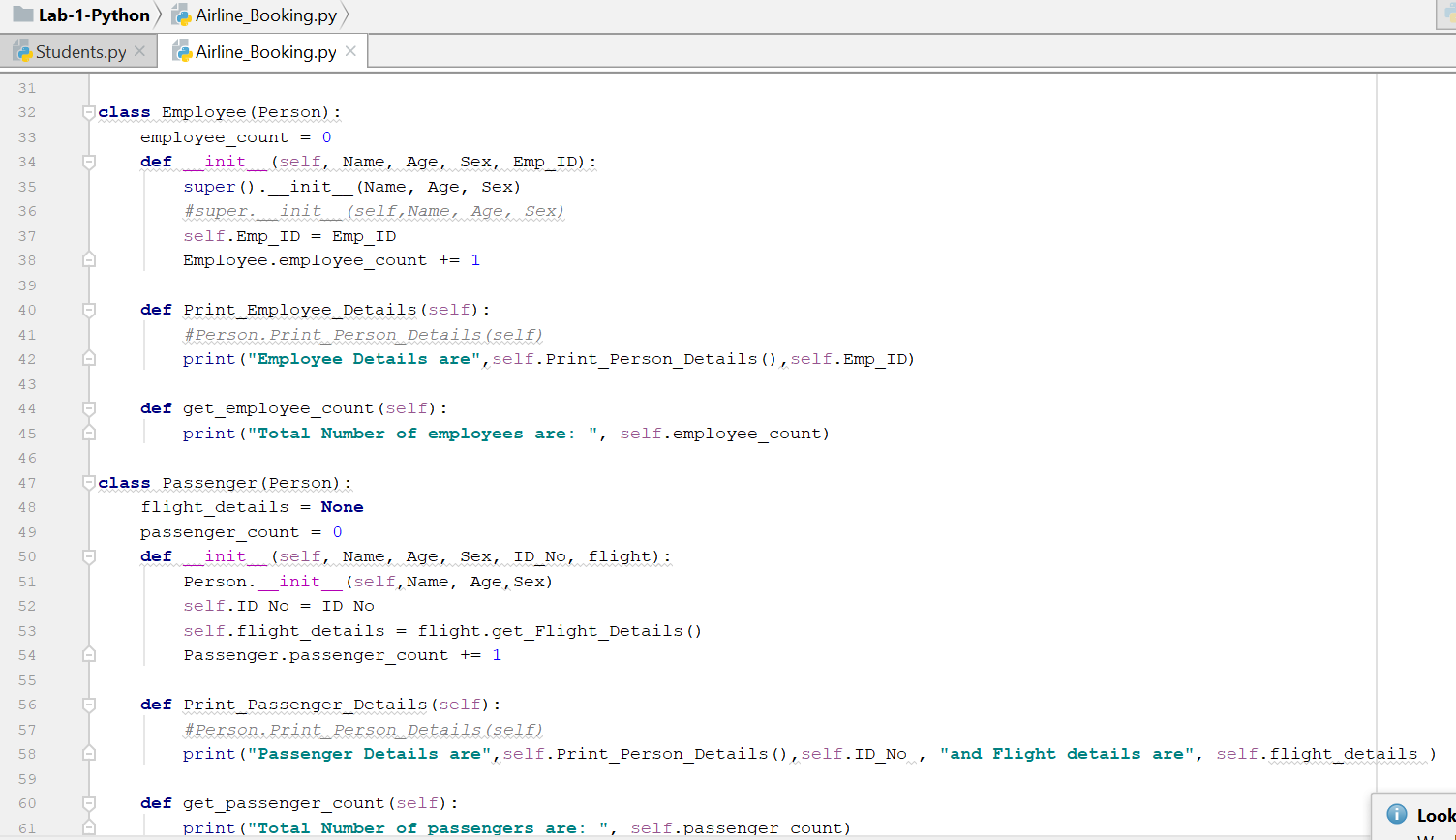
**Parameters:**

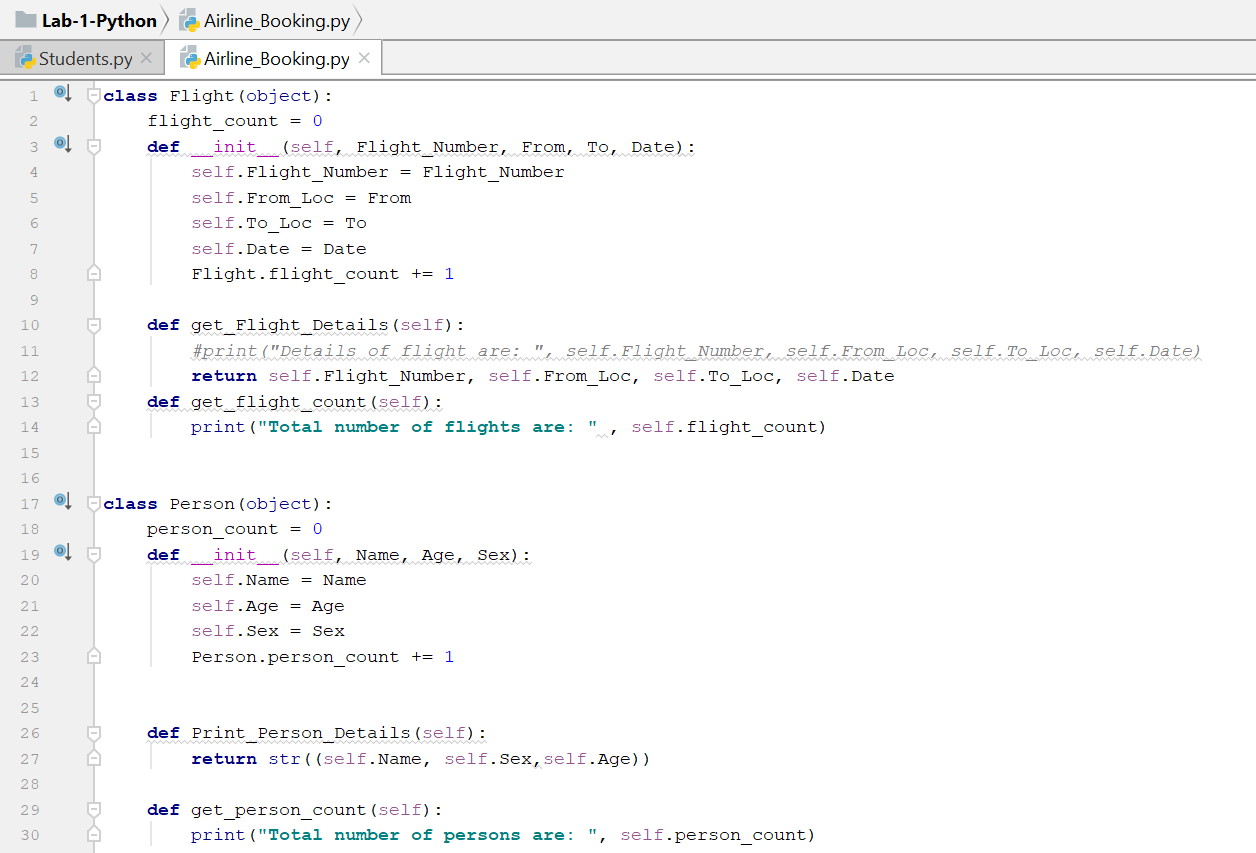
Creating the objects and calling the various methods implemented in the program.

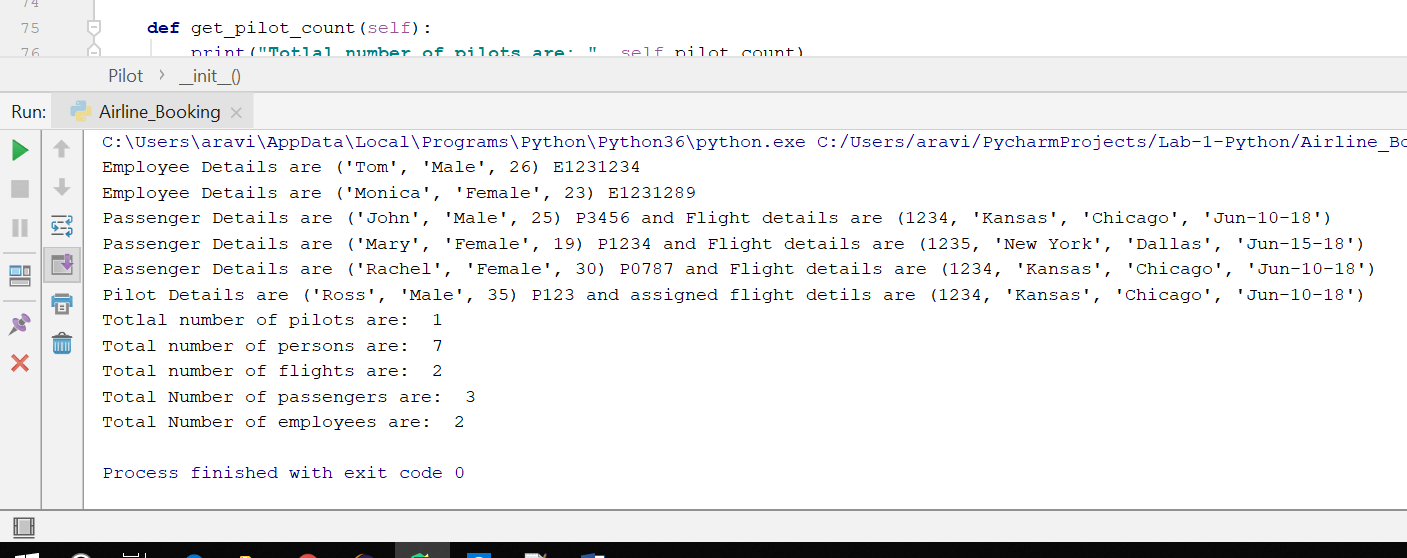
**Conclusion:**

The code worked fine, and it will print the details of every person, employee etc. and their count.

**Code and Output:**







**6.**

**Introduction:**

Input as NumPy Vector and finding the most frequent element in the array

**Objectives:**

Creating an NumPy vector and find the frequent element using NumPy bin count.

**Approach:**

Created a vector and used bincount method of NumPy and argmax.

**Workflow:**

Created a NumPy vector and found the most frequent element using the various methods of NumPy.

**Parameters:**

Creating the vector list and passing the list to find the frequent element.

**Conclusion:**

The code worked fine, and it will print the most frequent element.

**Code and Output:**

