- **1.WHAT IS PYTHON?**

Python is general purpose interpreted languagewith easy syntax & dynamic semantics

Created by Guido Van Rossum in 1989.

FEATURES OF PYTHON

1. Simplicity: Python has made programming fun because it's simple
2. Open Source: It is free for anyone to use. Modification in python is also allowed
3. Portability: Write your code and you can share your anyone u want.
4. Extendible & Extensible: Python supports adding code of other languages.
5. Interpreted: Management of memory, CPU and debugging the code is simpler
6. Huge Libraries: python has a huge library support which helps you in problem
7. Object Orientation: Python supports oops

Begginers Python Basics: From installation to Programming...

INSTALLATION OF PYTHON:

Just follow the following things to install python on your computer right.

Step 1: Just go to the browser and
type: https://www.python.org/downloads

Step 2: Click on download python 3.8.3 or any 3.x.x version you want

Step 3: Go to the download folder and double click on python file to install and follow the steps.

Step 4: Python 3 is installed on your computer. next step is we need to choose a text editor.

Step 5: To get the Text editor (Python IDE for developers) go to the https://www.jetbrains.com/pycharm

Step 6: There are two download versions professional and Community. just click on community version.

Step 7: Go to the download folder and double click on **PyCharm** file to install and follow the steps.

Step 8: This text editor help us while developing a code, it guide us the errors and all alright.

Step 9: Now python 3 and pycharm text editor both installed on computer and now we are ready to set up & Hello world Program.

How to create a First Python Program...

Step 1: First you have to open up the pycharm text editor.

Step 2: Just click on Create a New Project.

Step 3: In Location text box you have to specify the program name like FirstProg. **[C:\Users\Admin\PycharmProjects\FirstProg]**

Step 4: In Existing Interpreter you will see the our installed python3.8.3 interpreter is there. after that just click on Create button.

Step 5: Go to the FirstProg folder in right side right click on it and say New File ---> and select Python file.

Step 6: Give file name **app** so it will save the file in **app.py** format

Step 7: Start writing your first python program i:e **Hello world** using pycharmIDE.

Step 8: To run this program just click on **Run option** click on **Run** it will ask which file choose your file here the file is app so click on it and after that down below on **python console** you will see the output.

Step 9: That's all in this way you have to write your python code using PyCharm IDE.

Note: Still doing all of this if anyone is *facing problem in installation of both Python and PyCharm* they have to use this link for python coding but remember internet connection is must for this. i mean to say you have to use online Python Compiler(Interpreter) so without wasting a time just click on the link: https://www.programiz.com/python-programming/online-compiler/

How to create a Menu driven Program in Python

HOW TO CREATE A MENU DRIVEN PROGRAM IN PYTHON..

Step 1. Open your python command prompt.
Step 2. Write the following code into it.
Step 3. Execute it / Test the output for various inputs.
Step 4. Now move towards the study parts of python like how to declare and define function in python.
Step 5. To study function more in detail please click on the following link
Step 6. https://www.javatpoint.com/python-functions

The code to write in Step 2 are as follows:

```
(base) C:\Users\Admin>python
Python 3.7.6 (default, Jan  8 2020, 20:23:39) [MSC v.1916 64 bit
(AMD64)] :: Ana
```

```
conda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> def mainMenu():
...  print("1. Adition")
...  print("2. Substraction")
...  print("3. Exit")
...  choice=int(input("Enter u r choice"))
...  if choice==1:
...   add()
... elif choice==2:
  File "<stdin>", line 8
    elif choice==2:
       ^
SyntaxError: invalid syntax   Note: indentation of code is too much important in python otherwise you will get error like this.
 # Following is the code of main function in python.
>>> def mainMenu():
...  print("1. Adition")
...  print("2. Substraction")
...  print("3. Exit")
...  choice=int(input("Enter u r choice"))
...  if choice==1:
...   add() #when choice is equals to 1 we call add() function from mainMenu()
...  elif choice==2:
...   sub() #when choice is equals to 2 we call sub() function from mainMenu()
...  elif choice==3:
...   exit  #when choice is equals to 3 we call exit function from mainMenu()
...  else:
...   print("Invalid choice, Enter valid choice1-3")
...   mainMenu()
...# to come out from function press two times Enter key.
```

```
# To call mainMenu function for execution.
>>> mainMenu()
1. Adition
2. Substraction
3. Exit
Enter u r choice5   # here we are testing our code for invalid input.
Invalid choice, Enter valid choice1-3
1. Adition
2. Substraction
3. Exit
Enter u r choice3   # for valid inputs its works fine
```

```
# Definition of add() function that is actual logic of addition function
how it behaves after calling.
>>> def add():
...  print("Enter no1:")
...  x=int(input())
...  y=int(input("Enter no2:"))
...  z=x+y
...  print("The add is:",z)
...  mainMenu()
...
>>>  mainMenu()  # space is not allowed while calling function
  File "<stdin>", line 1
    mainMenu()
    ^
IndentationError: unexpected indent
>>> mainMenu()
1. Adition
2. Substraction
3. Exit
Enter u r choice1 #Execution or calling of sub() function
Enter no1:
2
Enter no2:2
The add is: 4
1. Adition
2. Substraction
3. Exit
Enter u r choice4
Invalid choice, Enter valid choice1-3
1. Adition
2. Substraction
3. Exit
Enter u r choice3
```

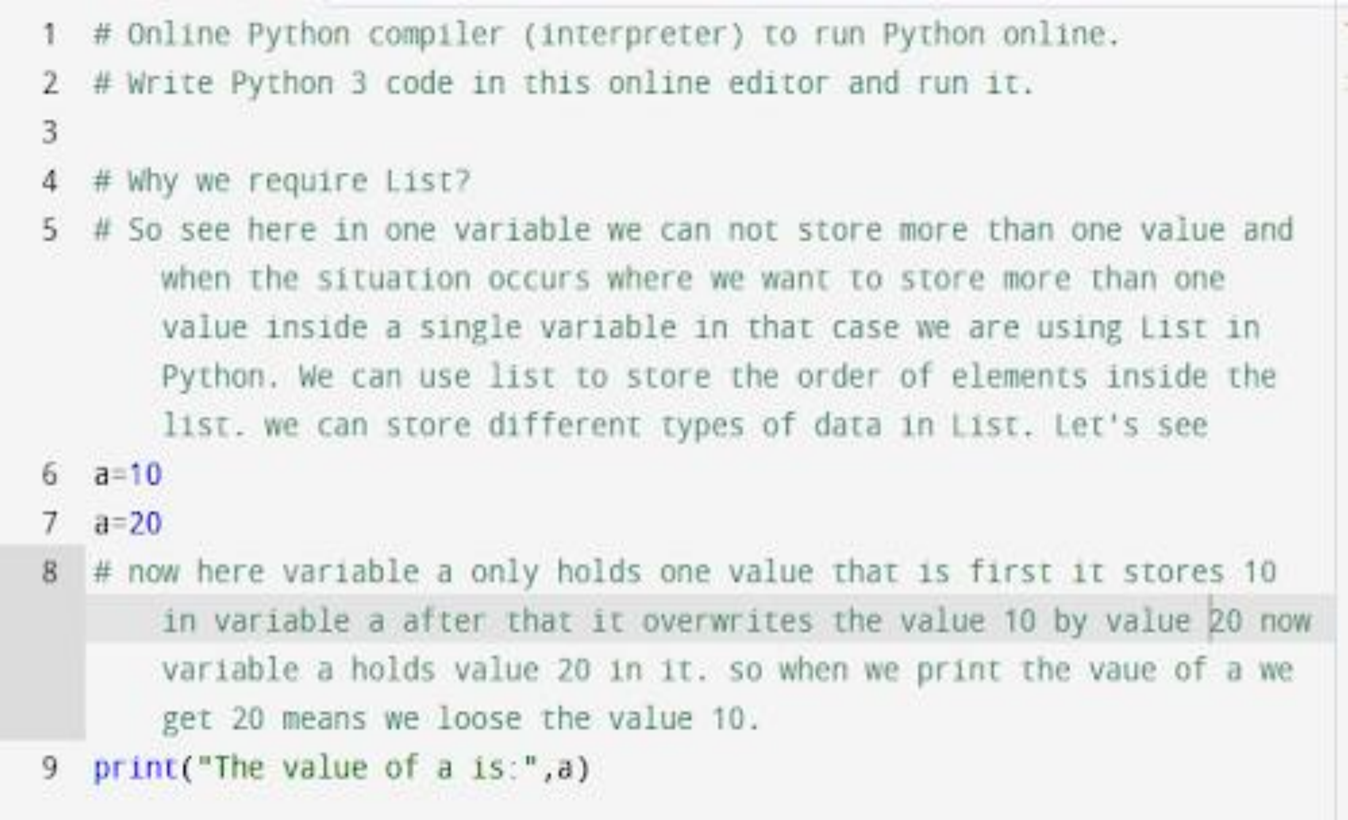
```
# Definition of sub() function that is actual logic of substraction function how it behaves after calling.
>>> def sub():
...  a=int(input("enter no1"))
...  b=int(input("enter no2"))
...  print("The sub is:",a-b)
...  mainMenu()
...
>>> mainMenu()
1. Adition
2. Substraction
3. Exit
Enter u r choice2 #Execution or calling of sub() function
enter no120
enter no210
The sub is: 10
1. Adition
2. Substraction
3. Exit
```

```
Enter u r choice3
>>>
```

What is list in python?

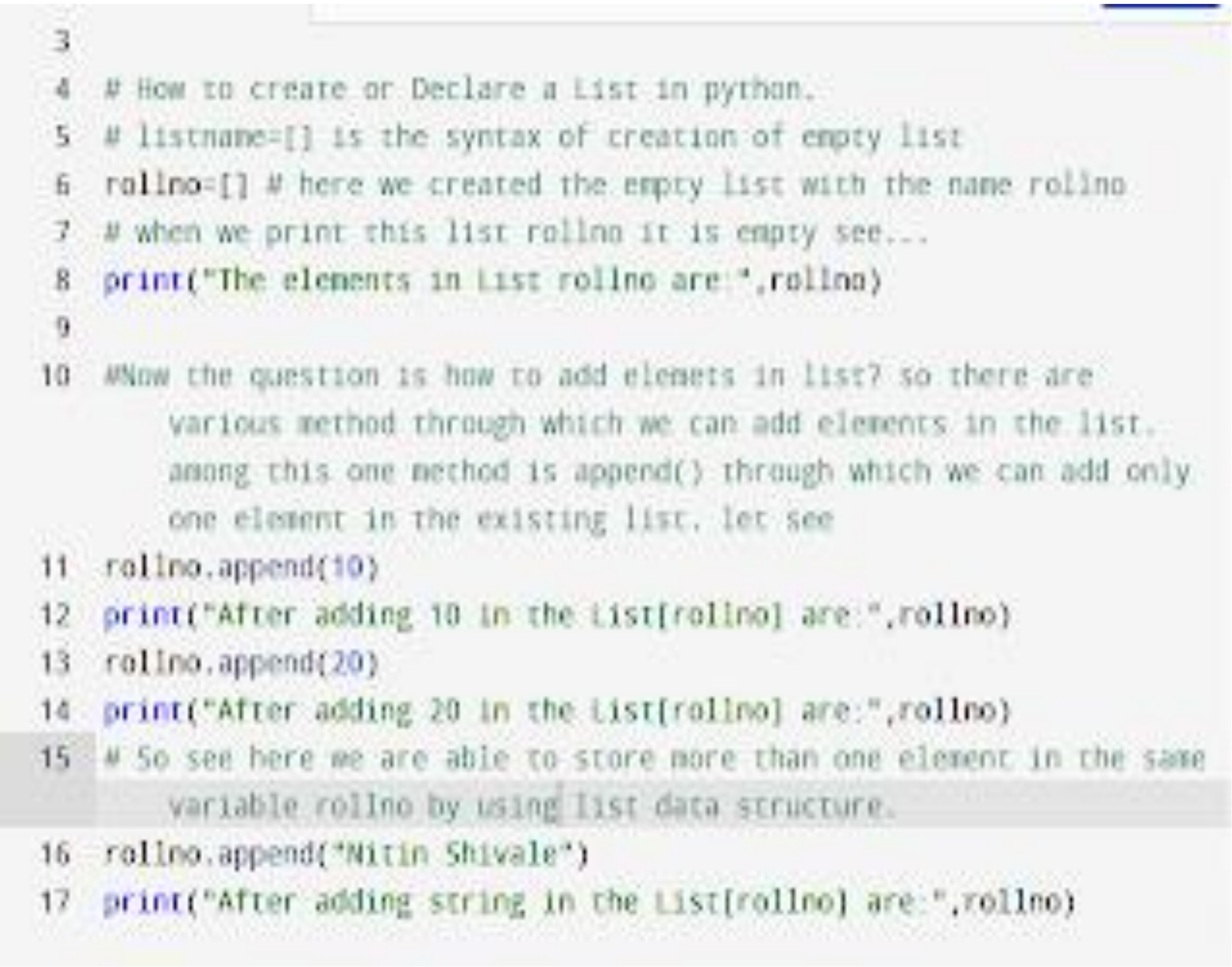
Lists are just like dynamic sized arrays.

Why we require list in python?

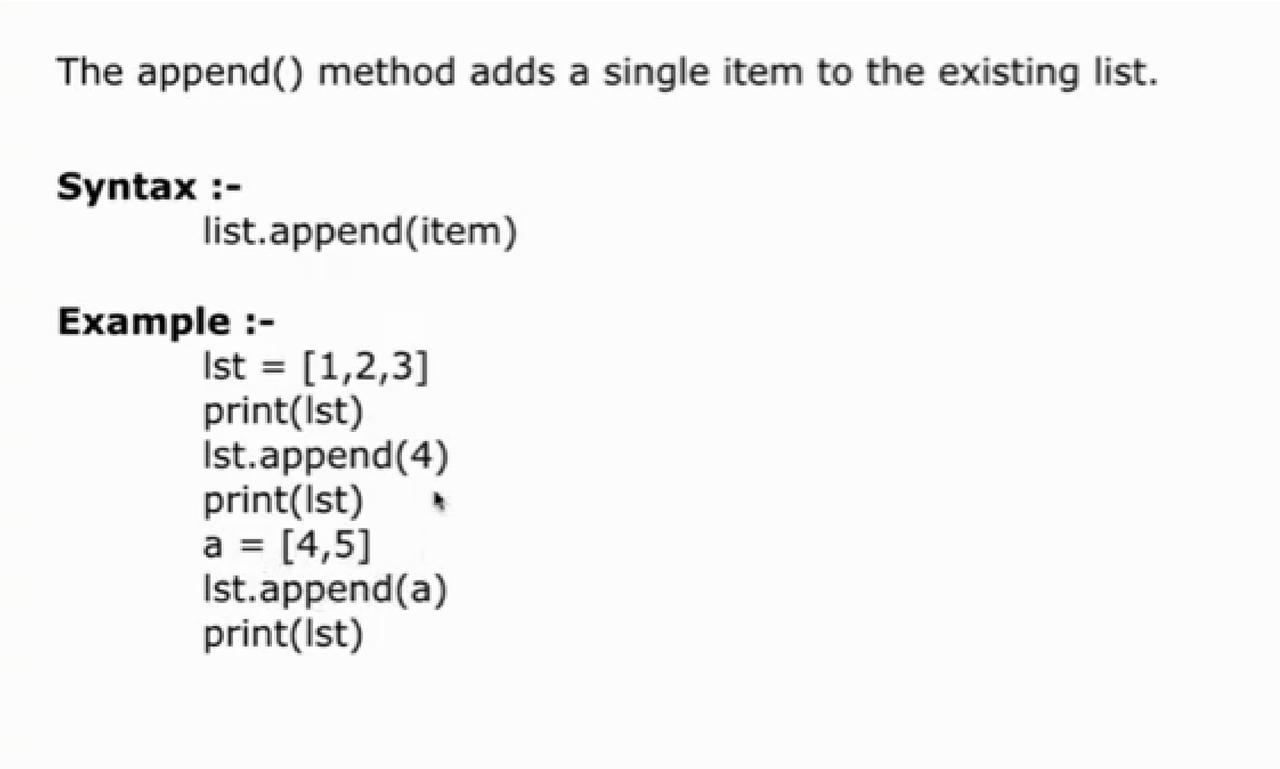


```
# Online Python compiler (interpreter) to run Python online.
# Write Python 3 code in this online editor and run it.

# Why we require List?
# So see here in one variable we can not store more than one value and when the situation occurs where we want to store more than one value inside a single variable in that case we are using List in Python. We can use list to store the order of elements inside the list. we can store different types of data in List. Let's see
a=10
a=20
# now here variable a only holds one value that is first it stores 10 in variable a after that it overwrites the value 10 by value 20 now variable a holds value 20 in it. so when we print the vaue of a we get 20 means we loose the value 10.
print("The value of a is:",a)
```

Lists need not be stored similar data types of elements always which makes it a most powerful tool in Python. A single list may contain different types of data like Float,Integers, Strings, as well as Objects.



```

# How to create or Declare a List in python.
# listname=[] is the syntax of creation of empty list
rollno=[] # here we created the empty list with the name rollno
# when we print this list rollno it is empty see...
print("The elements in List rollno are:",rollno)

#Now the question is how to add elemets in list? so there are various method through which we can add elements in the list. among this one method is append() through which we can add only one element in the existing list. let see
rollno.append(10)
print("After adding 10 in the List[rollno] are:",rollno)
rollno.append(20)
print("After adding 20 in the List[rollno] are:",rollno)
# So see here we are able to store more than one element in the same variable rollno by using list data structure.
rollno.append("Nitin Shivale")
print("After adding string in the List[rollno] are:",rollno)
```



The append() method adds a single item to the existing list.

Syntax :-

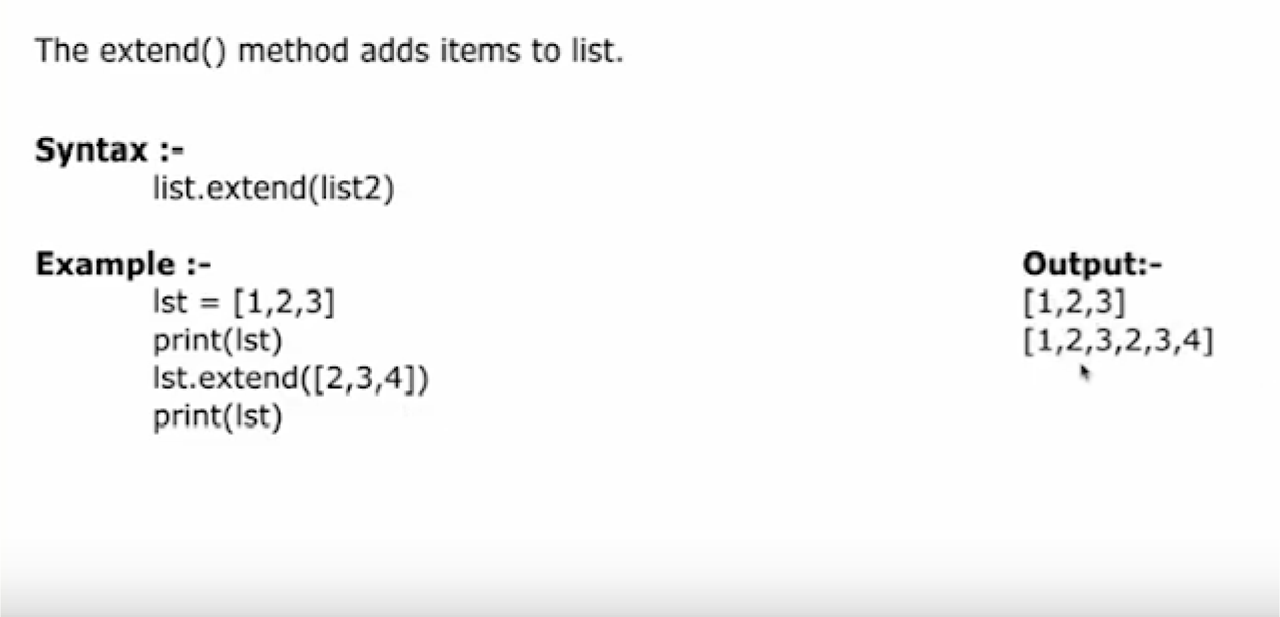
```
        list.append(item)
```

Example :-

```
        lst = [1,2,3]
        print(lst)
        lst.append(4)
        print(lst)
        a = [4,5]
        lst.append(a)
        print(lst)
```

Lists are mutable, and hence, they can be altered even after their creation.
List in Python are ordered and have a definite count. The elements in a list are indexed according to a definite sequence and the indexing of a list is done with 0 being the first index. Each element in the list has its definite place in the list, which allows duplicating of elements in the list, with each element having its own distinct place and credibility.
Note- Lists are a useful tool for preserving a sequence of data and further iterating over it.

MORE FUNCTION IN LIST:



The extend() method adds items to list.

Syntax :-

```
list.extend(list2)
```

Example :-

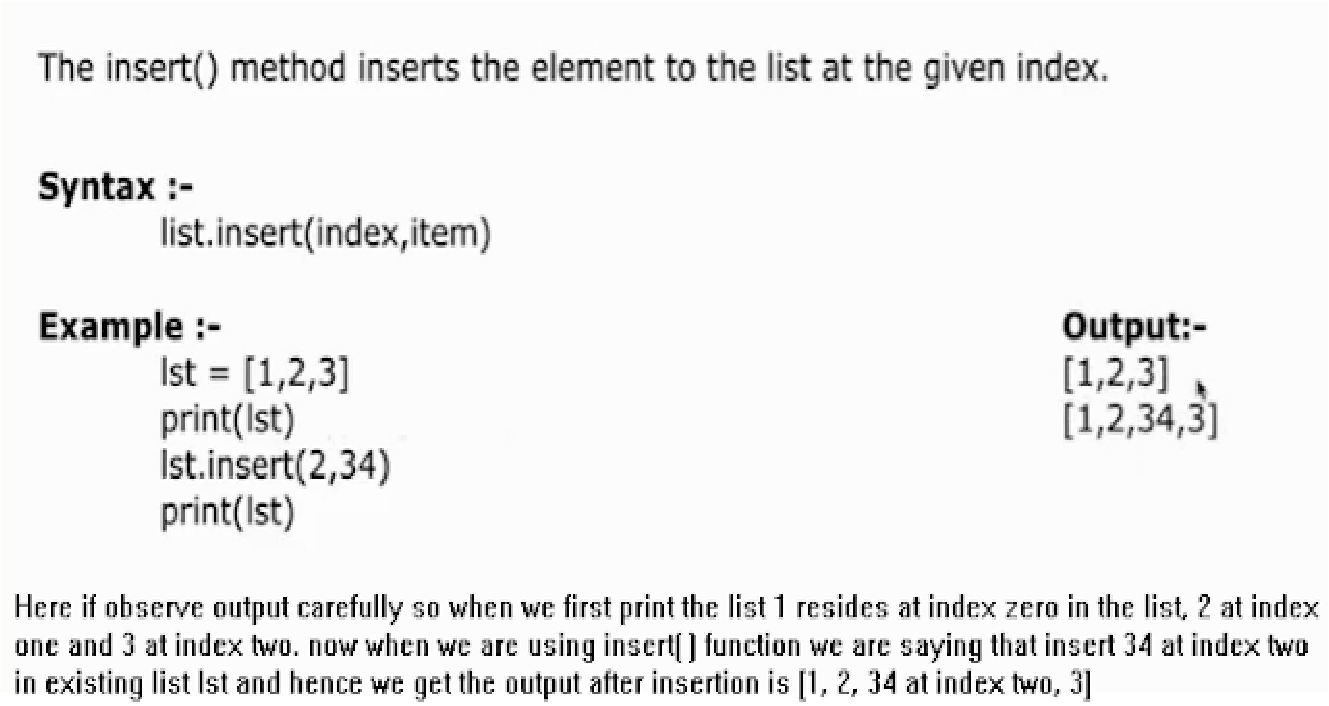
```
lst = [1,2,3]
print(lst)
lst.extend([2,3,4])
print(lst)
```

Output:-

```
[1,2,3]
[1,2,3,2,3,4]
```

Note: extend() simple adds the elements in existing list it doesn't make Nested list.

Next Function is insert( ) in Python



The insert() method inserts the element to the list at the given index.

Syntax :-

```
list.insert(index,item)
```

Example :-

```
lst = [1,2,3]
print(lst)
lst.insert(2,34)
print(lst)
```

Output:-

```
[1,2,3]
[1,2,34,3]
```

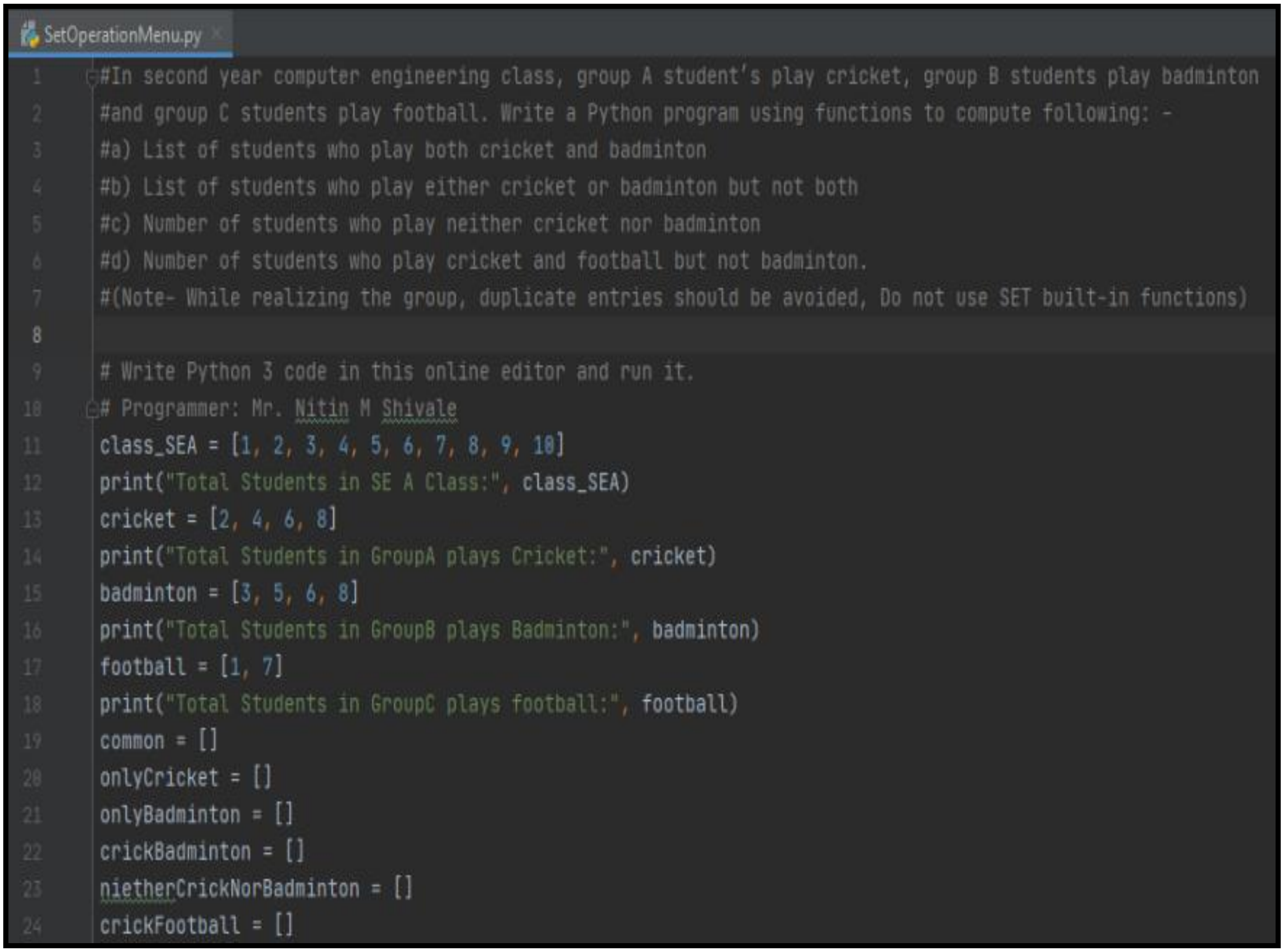
Here if observe output carefully so when we first print the list 1 resides at index zero in the list, 2 at index one and 3 at index two. now when we are using insert( ) function we are saying that insert 34 at index two in existing list lst and hence we get the output after insertion is [1, 2, 34 at index two, 3]

Problem Statement:

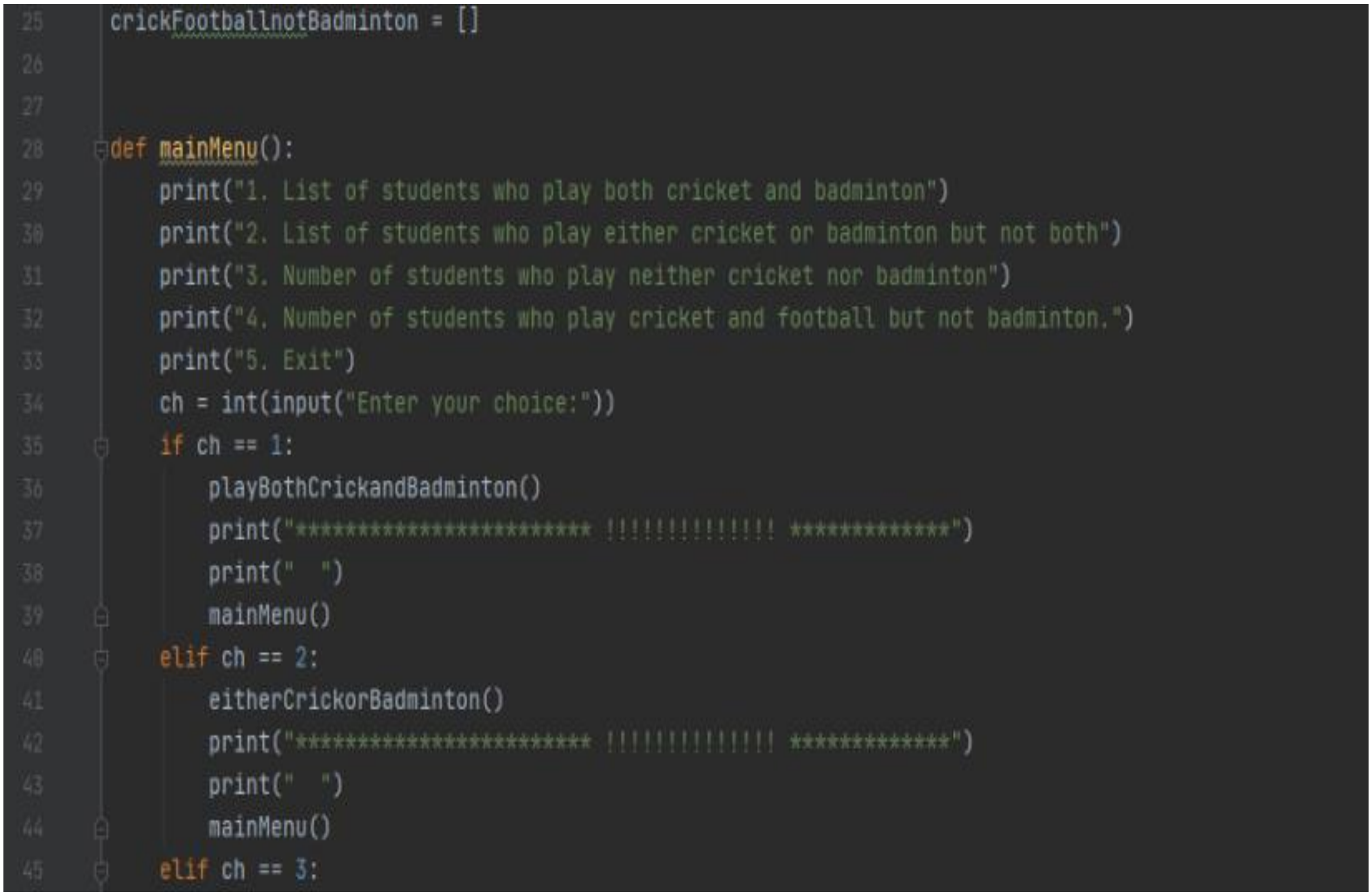
In second year computer engineering class, group A student’s play cricket, group B students play badminton and group C students play football. Write a Python program using functions to compute following: -

a) List of students who play both cricket and badminton
b) List of students who play either cricket or badminton but not both
c) Number of students who play neither cricket nor badminton
d) Number of students who play cricket and football but not badminton.
(Note- While realizing the group, duplicate entries should be avoided, Do not use SET built-in functions)

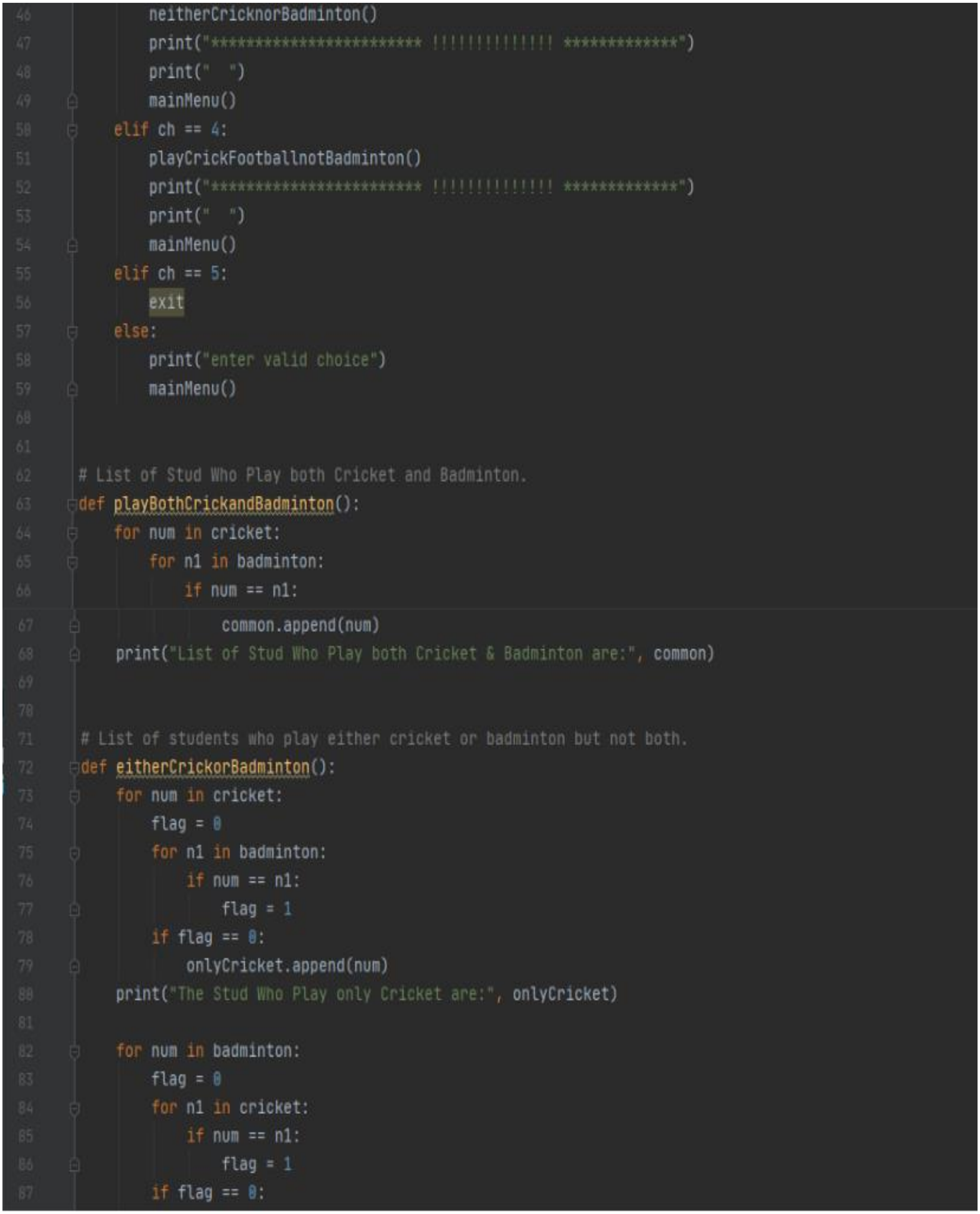
Source Program:



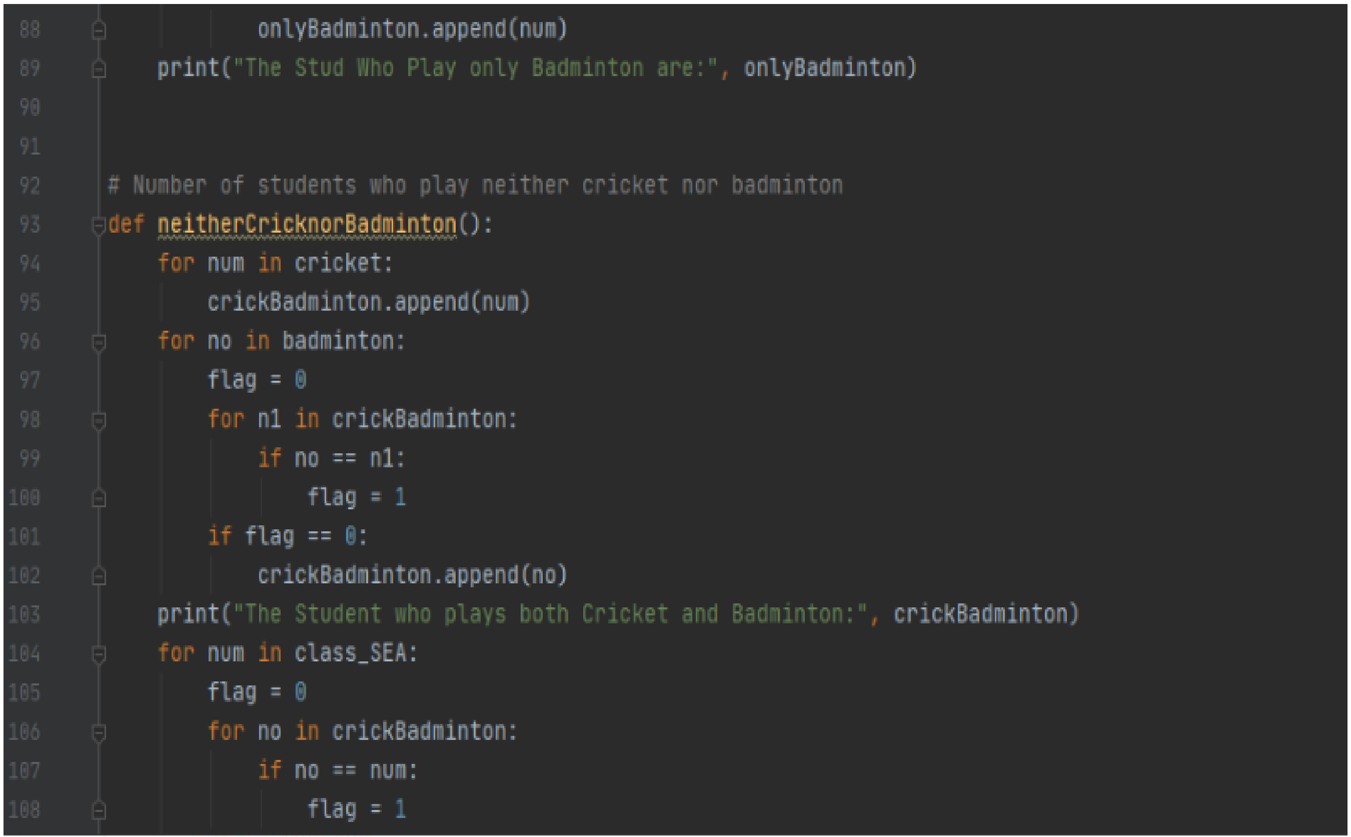
```
#In second year computer engineering class, group A student’s play cricket, group B students play badminton
#and group C students play football. Write a Python program using functions to compute following: -
#a) List of students who play both cricket and badminton
#b) List of students who play either cricket or badminton but not both
#c) Number of students who play neither cricket nor badminton
#d) Number of students who play cricket and football but not badminton.
#(Note- While realizing the group, duplicate entries should be avoided, Do not use SET built-in functions)

# Write Python 3 code in this online editor and run it.
# Programmer: Mr. Nitin M Shivale
class_SEA = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
print("Total Students in SE A Class:", class_SEA)
cricket = [2, 4, 6, 8]
print("Total Students in GroupA plays Cricket:", cricket)
badminton = [3, 5, 6, 8]
print("Total Students in GroupB plays Badminton:", badminton)
football = [1, 7]
print("Total Students in GroupC plays football:", football)
common = []
onlyCricket = []
onlyBadminton = []
crickBadminton = []
nietherCrickNorBadminton = []
crickFootball = []
```



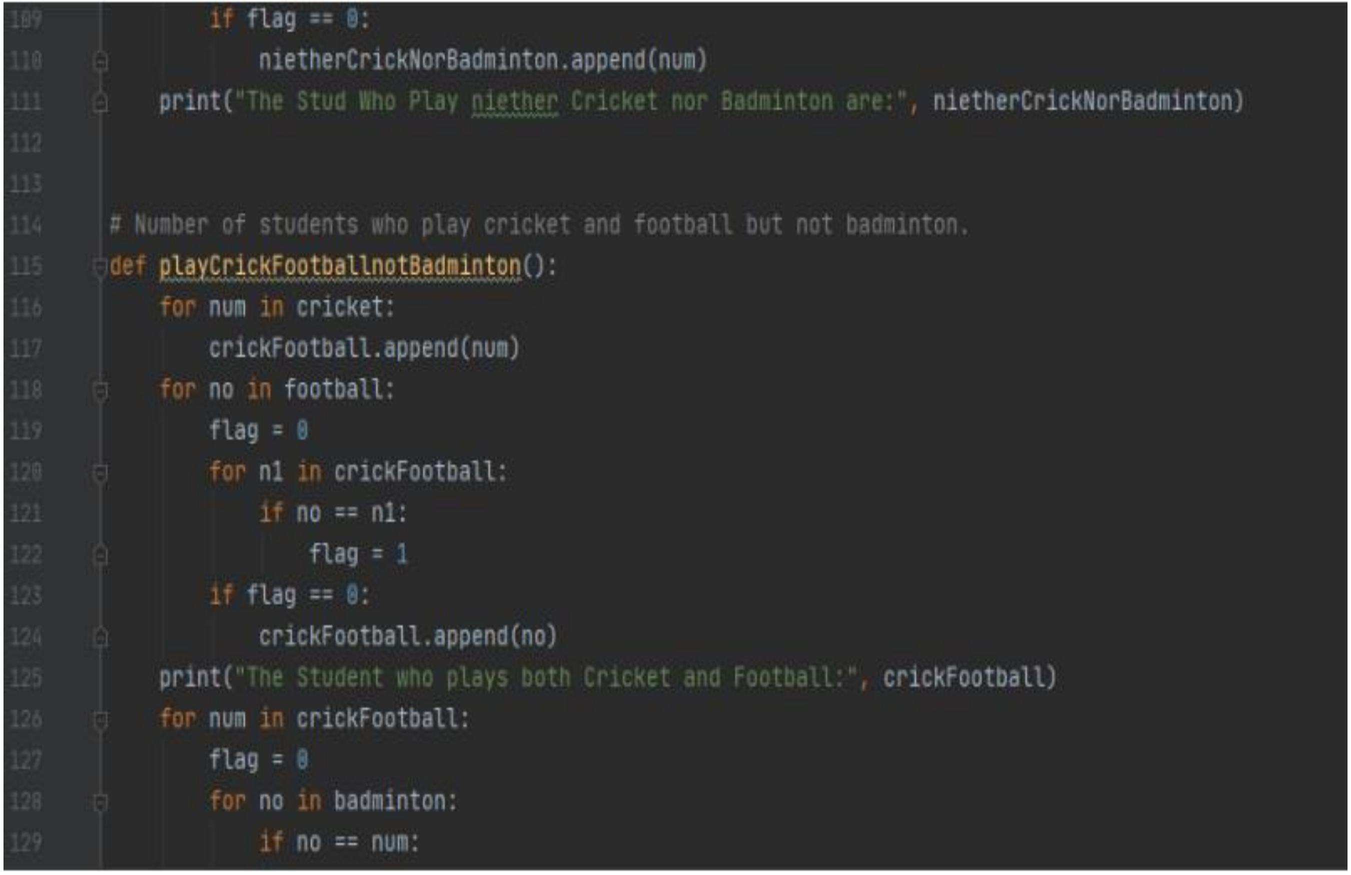
```
crickFootballnotBadminton = []


def mainMenu():
    print("1. List of students who play both cricket and badminton")
    print("2. List of students who play either cricket or badminton but not both")
    print("3. Number of students who play neither cricket nor badminton")
    print("4. Number of students who play cricket and football but not badminton.")
    print("5. Exit")
    ch = int(input("Enter your choice:"))
    if ch == 1:
        playBothCrickandBadminton()
        print("************************ !!!!!!!!!!!!!! *************")
        print("  ")
        mainMenu()
    elif ch == 2:
        eitherCrickorBadminton()
        print("************************ !!!!!!!!!!!!!! *************")
        print("  ")
        mainMenu()
    elif ch == 3:
```



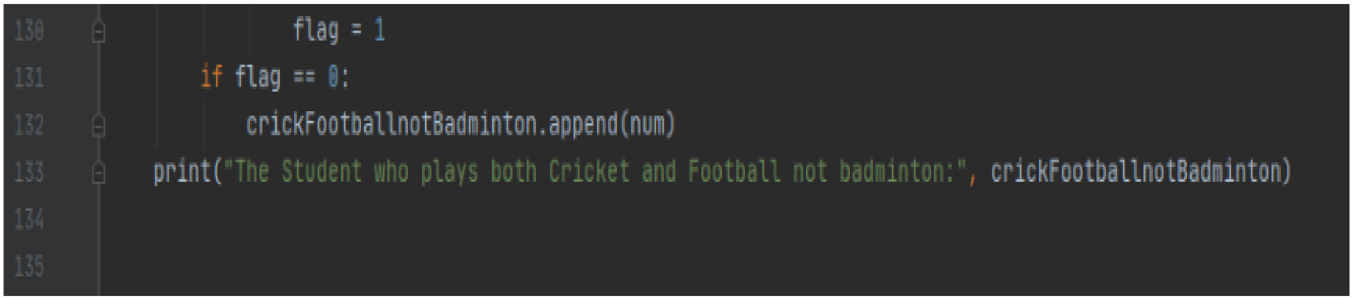
```python
        neitherCricknorBadminton()
        print("************************ !!!!!!!!!!!!!! *************")
        print("  ")
        mainMenu()
    elif ch == 4:
        playCrickFootballnotBadminton()
        print("************************ !!!!!!!!!!!!!! *************")
        print("  ")
        mainMenu()
    elif ch == 5:
        exit
    else:
        print("enter valid choice")
        mainMenu()


# List of Stud Who Play both Cricket and Badminton.
def playBothCrickandBadminton():
    for num in cricket:
        for n1 in badminton:
            if num == n1:
                common.append(num)
    print("List of Stud Who Play both Cricket & Badminton are:", common)


# List of students who play either cricket or badminton but not both.
def eitherCrickorBadminton():
    for num in cricket:
        flag = 0
        for n1 in badminton:
            if num == n1:
                flag = 1
        if flag == 0:
            onlyCricket.append(num)
    print("The Stud Who Play only Cricket are:", onlyCricket)

    for num in badminton:
        flag = 0
        for n1 in cricket:
            if num == n1:
                flag = 1
        if flag == 0:
```



```python
            onlyBadminton.append(num)
    print("The Stud Who Play only Badminton are:", onlyBadminton)


# Number of students who play neither cricket nor badminton
def neitherCricknorBadminton():
    for num in cricket:
        crickBadminton.append(num)
    for no in badminton:
        flag = 0
        for n1 in crickBadminton:
            if no == n1:
                flag = 1
        if flag == 0:
            crickBadminton.append(no)
    print("The Student who plays both Cricket and Badminton:", crickBadminton)
    for num in class_SEA:
        flag = 0
        for no in crickBadminton:
            if no == num:
                flag = 1
```



```python
        if flag == 0:
            nietherCrickNorBadminton.append(num)
    print("The Stud Who Play niether Cricket nor Badminton are:", nietherCrickNorBadminton)


# Number of students who play cricket and football but not badminton.
def playCrickFootballnotBadminton():
    for num in cricket:
        crickFootball.append(num)
    for no in football:
        flag = 0
        for n1 in crickFootball:
            if no == n1:
                flag = 1
        if flag == 0:
            crickFootball.append(no)
    print("The Student who plays both Cricket and Football:", crickFootball)
    for num in crickFootball:
        flag = 0
        for no in badminton:
            if no == num:
```



```
                flag = 1
        if flag == 0:
            crickFootballnotBadminton.append(num)
    print("The Student who plays both Cricket and Football not badminton:", crickFootballnotBadminton)
```

Output:

```
Total Students in SE A Class: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
Total Students in GoupA plays Cricket: [2, 4, 6, 8]
Total Students in GoupB plays Badminton: [3, 5, 6, 8]
Total Students in GoupC plays football: [1, 7]
>>> mainMenu()
1. List of students who play both cricket and badminton
2. List of students who play either cricket or badminton but not both
3. Number of students who play neither cricket nor badminton
4. Number of students who play cricket and football but not badminton.
5. Exit
Enter your choice:1
List of Stud Who Play both Cricket & Badminton are: [6, 8]
************************ !!!!!!!!!!!!!!
***************************

1. List of students who play both cricket and badminton
2. List of students who play either cricket or badminton but not both
3. Number of students who play neither cricket nor badminton
4. Number of students who play cricket and football but not badminton.
5. Exit
```

```
Enter your choice:2
The Stud Who Play only Cricket are: [2, 4]
The Stud Who Play only Badminton are: [3, 5]
************************ !!!!!!!!!!!!!
***************************

1. List of students who play both cricket and badminton
2. List of students who play either cricket or badminton but not both
3. Number of students who play neither cricket nor badminton
4. Number of students who play cricket and football but not badminton.
5. Exit
Enter your choice:3
The Student who plays both Cricket and Badminton: [2, 4, 6, 8, 3, 5]
The Stud Who Play neither Cricket nor Badminton are: [1, 7, 9, 10]
************************ !!!!!!!!!!!!!
***************************

1. List of students who play both cricket and badminton
2. List of students who play either cricket or badminton but not both
3. Number of students who play neither cricket nor badminton
4. Number of students who play cricket and football but not badminton.
5. Exit
Enter your choice:4
The Student who plays both Cricket and Football: [2, 4, 6, 8, 1, 7]
The Student who plays both Cricket and Football not badminton: [2, 4,
1, 7]
************************ !!!!!!!!!!!!!
***************************

1. List of students who play both cricket and badminton
```

```
2. List of students who play either cricket or badminton but not both
3. Number of students who play neither cricket nor badminton
4. Number of students who play cricket and football but not badminton.
5. Exit
Enter your choice:5
```

First of all we have to analyze the problem statement, we have to find out the following things from the Problem statement.

1. What is given in the Problem statement? [Given]

2. Identify the entity and it's attributes. [Entity]

3. Identify the users actual requirements and what are the inputs we are accepting from user.[Input]

4. Identify to store and process the data which data structure is required? [Data Structure]

5. How many functions are there in Problem statement? [Output]

<u>Analysis of Problem Statement by referring above five points:</u>

1. Given:

In the above Problem Statement **Given/Assume** things are:

a. **Class SEA Student Roll numbers**. that is whole class strength of SEA

b. Students Plays in **Group_A** Cricket: **Group_B** Badminton: and **Group_C** Football:

2. Entity:

Here the **entity** is **Class_SE_A** and the **attribute** is **Rollno**.

3. Input:

As a **input** we have to accept following things from the user:

3.1 Group_A Cricket: The Rollno of students who plays Cricket from Class_SE_A.

3.2 Group_B Badminton: The Rollno of students who plays Badminton from Class_SE_A.

3.3 Group_C Football: The Rollno of students who plays Football from Class_SE_A.

3.4 Class_SE_A: we have to assume the N number of students in class SE A.

4. Data Structure:

To **Store and Process the data as per user requirement** we are using **LIST** as a data Structure in Python Programming Language.

5. Output:

As per the user requirement **user wants following things as an output** when he is going to process the data.

5.1 List of students who play both cricket and badminton.

5.2 List of students who play either cricket or badminton but not both.

5.3 Number of students who play neither cricket nor badminton.

5.4 Number of students who play cricket and football but not badminton.

In this way we are going to analyze each and every problem statement before designing the algorithm that is called as **"Requirement Analysis"** in **S**oftware **D**evelopment **L**ife **C**ycle.

So by referring the above analysis,

we comes to know that we have to design **Four algorithms** for each and every Output.

Algorithm: For list of students who play both Cricket & Badminton.

```
Step 1: Start
Step 2: Accept group_A(Cricket) & group_B(Badminton) students roll nos in cricket
        and badminton set.
Step 3: Take roll no x from cricket set till roll no is avail, if not goto step 4
    Step 3.1 : Take roll no y from badminton set till roll no is avail, if not goto step 3
    Step 3.2:  Check if x is equals to y then store x in common set and go to step 3.1
    Step 3.3: if x is not equals to y then go to the step 3.1
Step 4: Display Common roll no from common set i:e the students roll no who
        plays both cricket and badminton
Step 5: Stop
```

Algorithm: For list of students who play either Cricket or Badminton.

```
Step 1: Start
Step 2: Accept group_A(Cricket) & group_B(Badminton) students roll nos in cricket
        and badminton set.
Step 3: Take roll no x from cricket set till roll no is avail, if not goto step 5
    Step 3.1 : set Flag value to zero i:e Flag=0
    Step 3.2: Take roll no y from badminton set till roll no is avail, if not goto step 4
    Step 3.3: Check if x is equals to y then set Flag=1 and go to the step 3.2, else if
              x is not equals to y then goto step 3.2
Step 4: check if Flag is equals to Zero i:e Flag==0 then store x in onlycricket set and
        goto step 3, else if Flag not equals to zero then directly goto step 3.
Step 5: Show the list of students who play only cricket from onlycricket set.
Step 6: Stop
```

Algorithm:Number of students who play neither cricket nor badminton.

```
Step 1: Start
Step 2: Accept group_A(Cricket) , group_B(Badminton) & whole class students roll nos in cricket, badminton & class_SEA sets.
Step 3: Take roll no x from cricket set till roll no is avail, and put it into cricketbadminton set.
Step 4: Take roll no y from badminton set till roll no is avail, if not goto step 6
    Step 4.1: set Flag value to zero i:e Flag=0
    Step 4.2: Take roll no x from cricketbadminton set till roll no is avail, if not goto step 5
    Step 4.3: Check if x is equals to y then set Flag=1 and go to the step 4.2, else if x is not equals to y then goto step 4.2
Step 5: check if Flag is equals to Zero i:e Flag==0 then store y in cricketbadminton set and goto step 4, else if Flag not equals to zero then directly goto step 4.
Step 6: Show the list of students who play both cricket and badminton from cricketbadminton set.
Step 7: Take roll no m1 from class_SEA set till roll no is avail, if not goto step 9
    Step 7.1: set Flag value to zero i:e Flag=0
    Step 7.2: Take roll no n1 from cricketbadminton set till roll no is avail, if not goto step 8
    Step 7.3: Check if m1 is equals to n1 then set Flag=1 and go to the step 7.2, else if m1 is not equals to n1 then goto step 7.2
Step 8: check if Flag is equals to Zero i:e Flag==0 then store m1 in neithercricketnorbadminton set and goto step 7, else if Flag not equals to zero then directly goto step 7.
Step 9: Show the list of students who play neither cricket nor badminton from neithercricketnorbadminton set.
Step 10; Stop
```

Algorithm:Number of students who play cricket and football but not badminton.

```
Step 1: Start

Step 2: Accept group_A(Cricket) , group_B(Badminton) & group_C(Football) students roll nos in cricket, badminton  & Football..

Step 3: Take roll no x from cricket set till roll no is avail, and put it into cricketfootball set.

Step 4:  Take roll no y from football set till roll no is avail, if not goto step 6
        Step 4.1: set Flag value to zero i:e Flag=0
        Step 4.2: Take roll no x from cricketfootball set till roll no is avail, if not goto step 5

        Step 4.3: Check if x is equals to y then set Flag=1 and go to the step 4.2, else if

                  x is not equals to y then goto step 4.2

Step 5: check if Flag is equals to Zero i:e Flag==0 then store y in cricketfootball set and goto step 4, else if Flag not
         equals to zero then directly goto step 4.
Step 6: Show the list of students who play both cricket and football  from cricketfootball set.

Step 7:  Take roll no m1 from badminton set till roll no is avail, if not goto step 9
        Step 7.1: set Flag value to zero i:e Flag=0
        Step 7.2: Take roll no n1 from cricketfootball set till roll no is avail, if not goto step 8

        Step 7.3: Check if m1 is equals to n1 then set Flag=1 and go to the step 7.2, else if m1 is not equals to n1 then goto Step 7.2

Step 8: check if Flag is equals to Zero i:e Flag==0 then store m1 in cricketfootballnotbadminton set and goto step 7, else if
         Flag not equals to zero then directly goto step 7.
Step 9: Show the list of students who play neither cricket nor badminton  from cricketfootballnotbadminton set.

Step 10; Stop
```