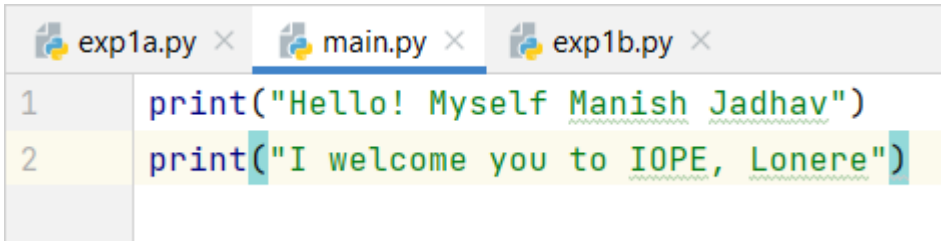
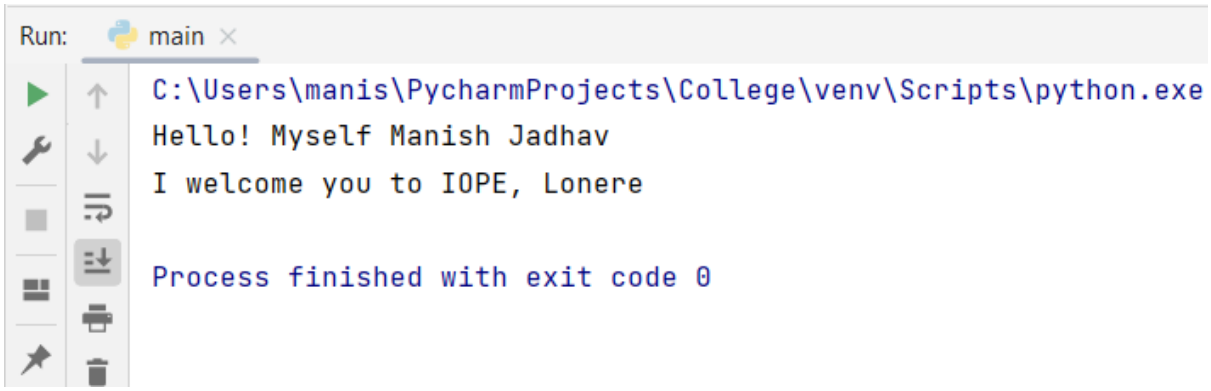
Experiment No.1

Aim: Edit/Compile/run a program to display the statements on two different lines.

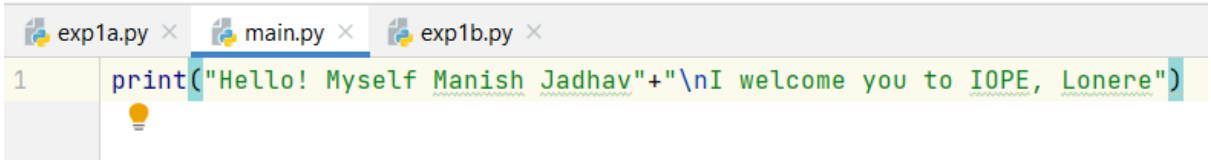
Program A:



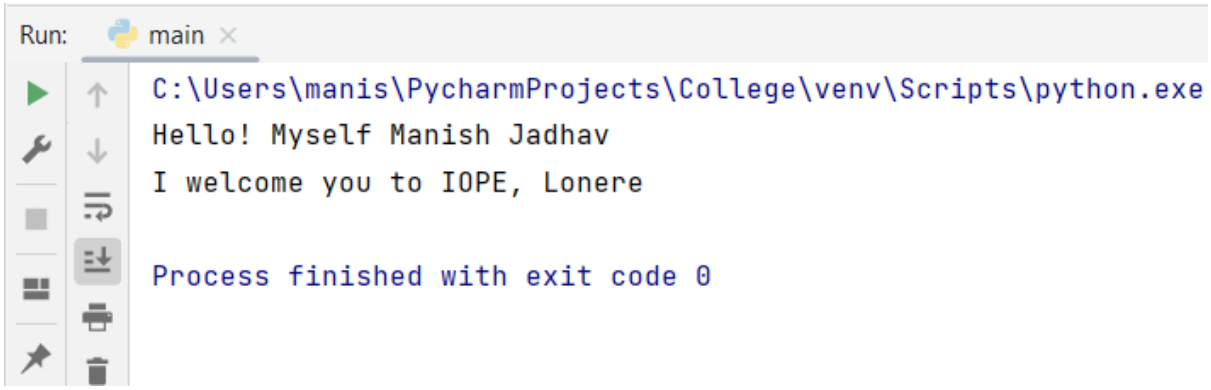
Output:



Program B:



Output:



Program C:

exp1a.py | main.py | exp1b.py

```
1  print("Best of India")
2  a="Maharashtra has "
3  b="Gujrat has "
4  c="Uttar Pradesh has "
5  d="Tamil Nadu has "
6  e="Karnatak has "
7  f="Uttarakhand has "
8  g="Odisha has "
9  print(a +"Raigad")
10 print(b +"Dwarka")
11 print(c +"Ayodhya")
12 print(d +"Ram Setu")
13 print(e +"Kishkinda")
14 print(f +"Kedarnath")
15 print(g +"Jagannath Puri Mandir")
```

Output:

Run: exp1a

```
C:\Users\manis\PycharmProjects\College\venv\Scripts\python.exe
Best of India
Maharashtra has Raigad
Gujrat has Dwarka
Uttar Pradesh has Ayodhya
Tamil Nadu has Ram Setu
Karnatak has Kishkinda
Uttarakhand has Kedarnath
Odisha has Jagannath Puri Mandir

Process finished with exit code 0
```

Program D:

exp1a.py | exp1b.py

```
1  print("TopMost Gaming Industries are as follows")
2  print("S8UL Esports"+"\nGodlike Esports"+"\nGlobal Esports"+"\nOrangutan Esports"
3     +"\nTeam XO"+"\nTeam IND"+"\nTSM Entity"+"\nTeam XSpark"+"\n7Sea Esports")
```

Output:

Run: exp1b

```
C:\Users\manis\PycharmProjects\College\venv\Scripts\python.exe
TopMost Gaming Industries are as follows
S8UL Esports
Godlike Esports
Global Esports
Orangutan Esports
Team XO
Team IND
TSM Entity
Team XSpark
7Sea Esports

Process finished with exit code 0
```

Writeup & Oral (4)	Practical Performance (4)	Attendance (2)	Total (10)

Experiment No.2

Aim: Edit/compile/run a program to display the statements on two different lines.

Program A:

exp2a.py

```
print('Hello World!')
print('Welcome to Python Programming!')
print("""This is the strangest
    way to print over
        multiple lines I know""")

```

Output:

Run: exp2a

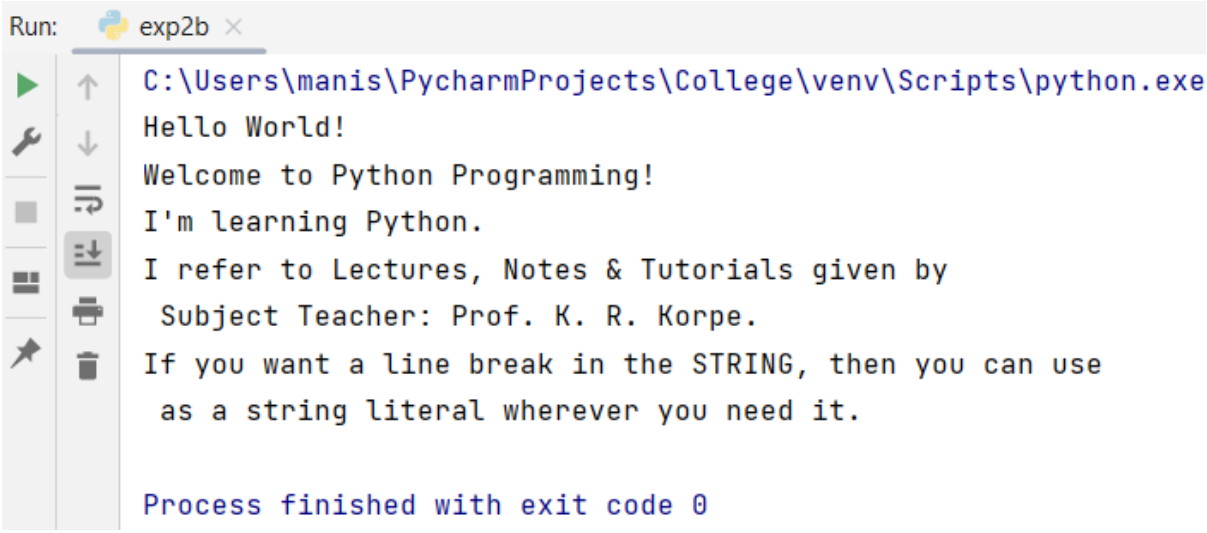
```
C:\Users\manis\PycharmProjects\College\venv\Scripts\python.exe
Hello World!
Welcome to Python Programming!
This is the strangest
    way to print over
        multiple lines I know

Process finished with exit code 0
```

Program B:

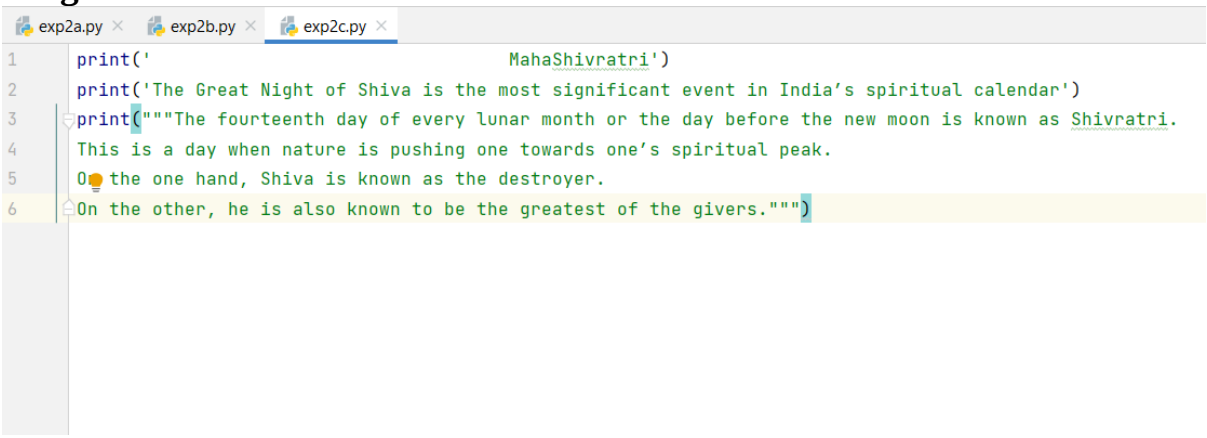
exp2a.py | exp2b.py

```
multiline_str = ("Hello World! \n"
"Welcome to Python Programming! \n"
"I'm learning Python.\n"
"I refer to Lectures, Notes & Tutorials given by \n Subject Teacher: Prof. K. R. Korpe.\n"
"If you want a line break in the STRING, then you can use" "\n" " as a string literal wherever you need it.")
print(multiline_str)
```

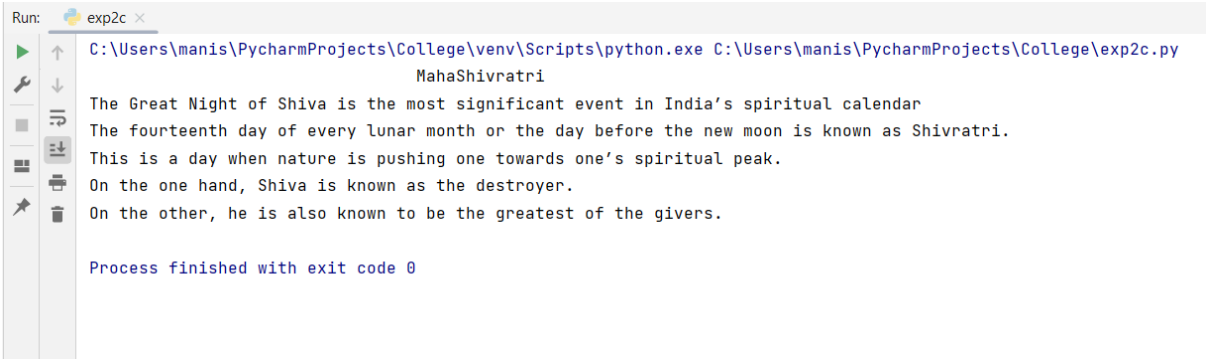
Output:



Program C:



Output:



Program D:

exp2a.py exp2b.py exp2c.py exp2d.py

```
multiline_str=("         Shiv Jayanti\n"
   "Shiv Jayanti is the birth anniversary of great Maratha ruler Shivaji Maharaj.\n"
      "Shivaji Maharaj was born on February 19, 1630, in Shivneri Fort.\n"
      "At the young age of 16, Shivaji Maharaj seized the Torna fort and \n by the age of 17 had seized the Raigad and"
            "Kondana forts.\n"
      "Shivaji Maharaj is considered as the greatest Maratha ruler.\n"
      )
print(multiline_str)
```

Output:

Run: exp2d

```
C:\Users\manis\PycharmProjects\College\venv\Scripts\python.exe C:\Users\manis\Pych
         Shiv Jayanti
Shiv Jayanti is the birth anniversary of great Maratha ruler Shivaji Maharaj.
Shivaji Maharaj was born on February 19, 1630, in Shivneri Fort.
At the young age of 16, Shivaji Maharaj seized the Torna fort and
 by the age of 17 had seized the Raigad and Kondana forts
Besides study I love to play sports
Such as Cricket, Kabaddi!


Process finished with exit code 0
```

Practical Performance (4)	Writeup & Oral (4)	Attendance (2)	Total (10)

Experiment No.3

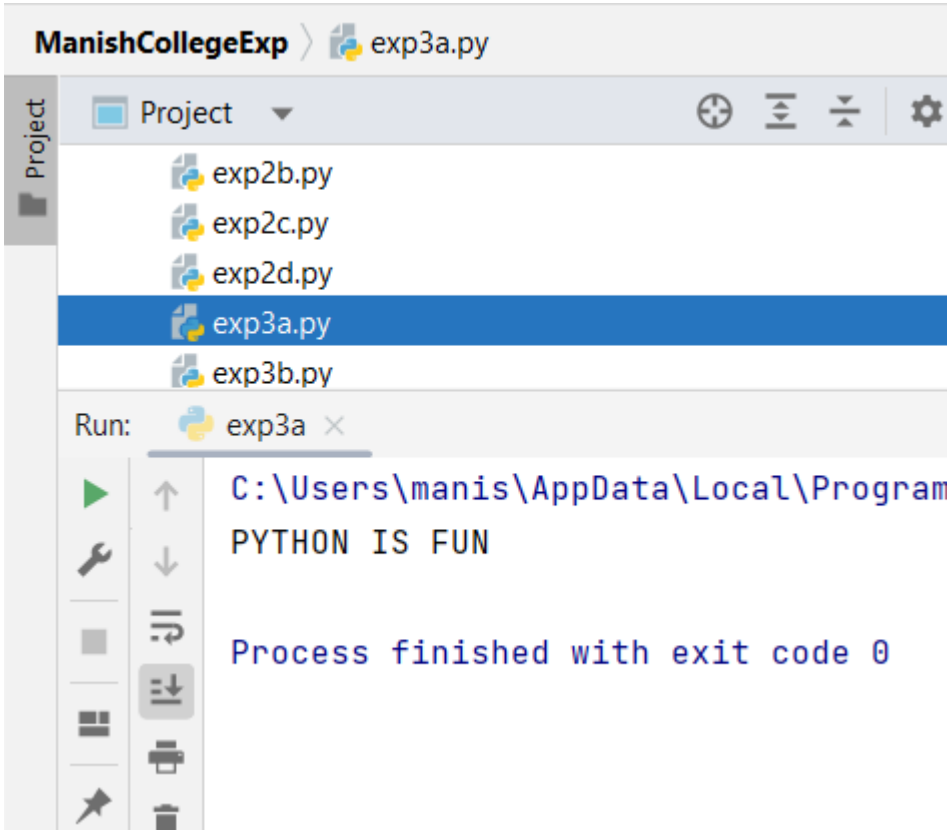
Aim: Edit/compile/run a program to initialize the string “hello world!” to a variable Str1 and convert the string into uppercase.

Program A:

exp3a.py

```
1  message='Python is fun'
2  print(message.upper())
3
```

Output:

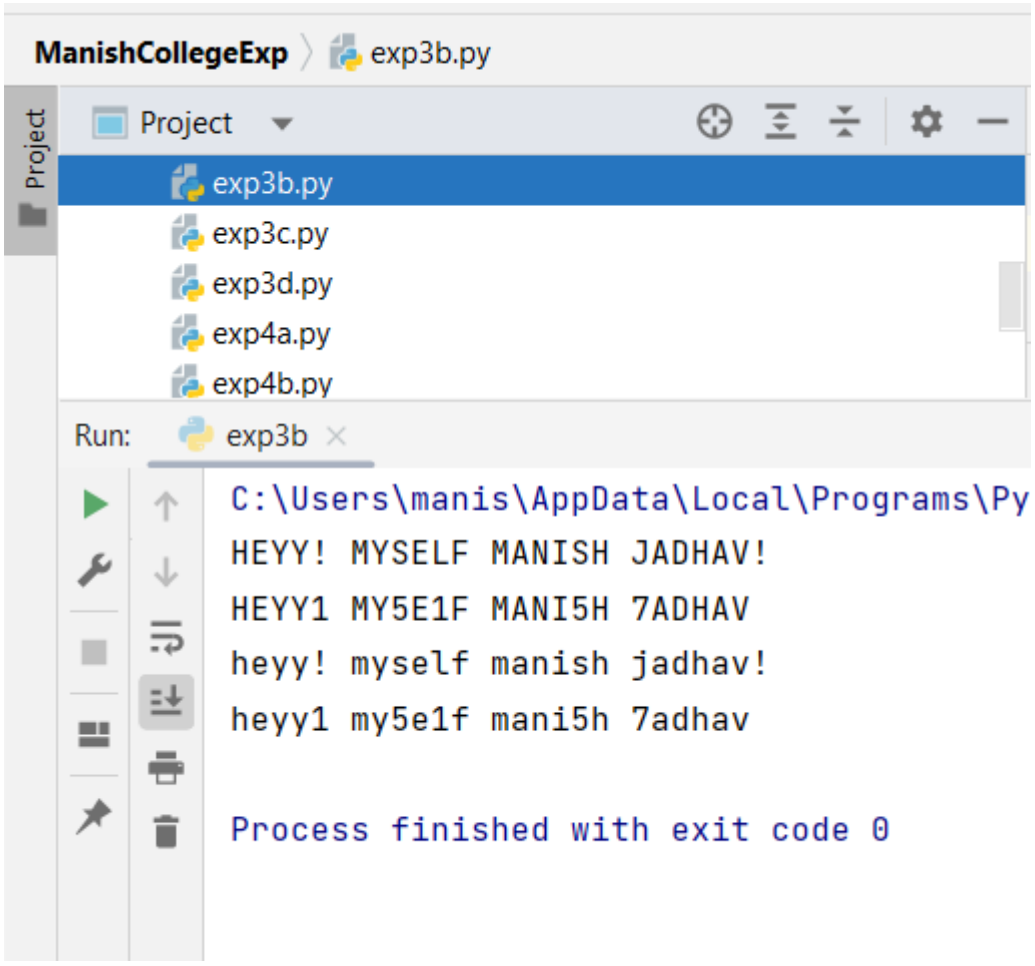


Program B:

exp3a.py × exp3b.py ×

```
1  string = "heyy! myself manish jadhav!"
2  print(string.upper())
3  string = "heyy1 my5e1f mani5h 7adhav"
4  print(string.upper())
5
6  string = "HEYY! MYSELF MANISH JADHAV!"
7  print(string.lower())
8  string = "HEYY1 MY5E1F MANI5H 7ADHAV"
9  print(string.lower())
```

Output:

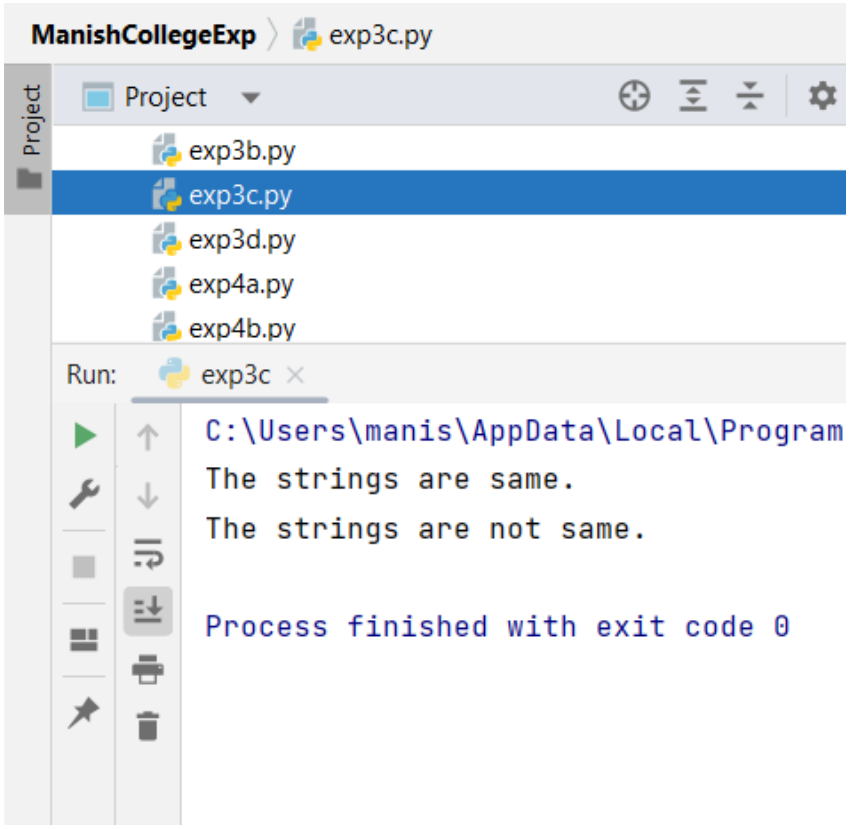


Program C:

exp3a.py | exp3b.py | exp3c.py | exp3d.py

```
1  firstString = "heyy! myself manish jadhav!"
2  secondString = "hEYy! mySelF maNIsH jadHAV!"
3  if(firstString.upper() == secondString.upper()):
4   print("The strings are same.")
5  else:
6   print("The strings are not same.")
7
8  firstString1 = "heyy! myself manish j adhav!"
9  secondString1 = "hEYy! mySelF maNIsH jadHAV!"
10 if(firstString1.upper() == secondString1.upper()):
11  print("The strings are same.")
12 else:
13  print("The strings are not same.")
14
```

Output:

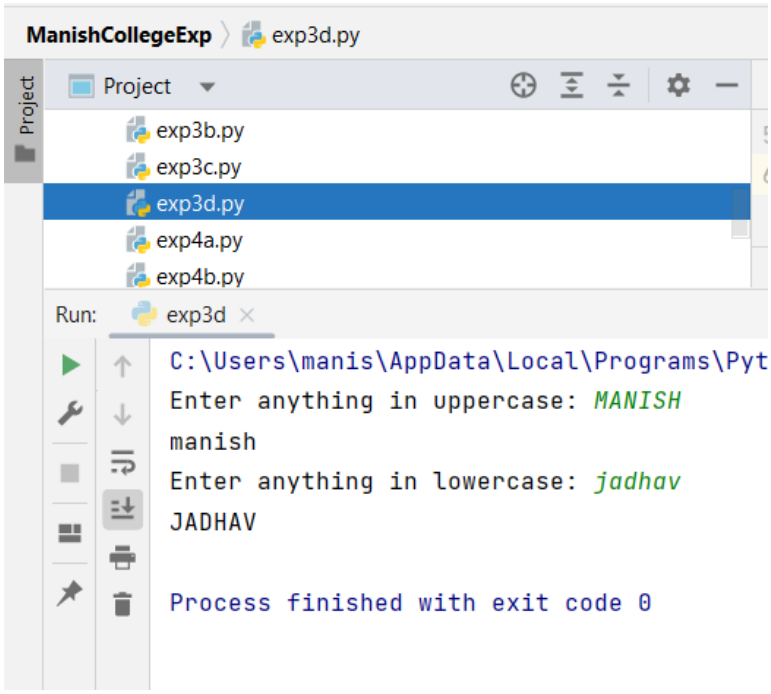


Program D:

exp3a.py | exp3b.py | exp3c.py | exp3d.py

```
1  msg=input("Enter anything in uppercase: ")
2  a=msg
3  print(a.swapcase())
4  str=input("Enter anything in lowercase: ")
5  b=str
6  print(b.swapcase())
```

Output:

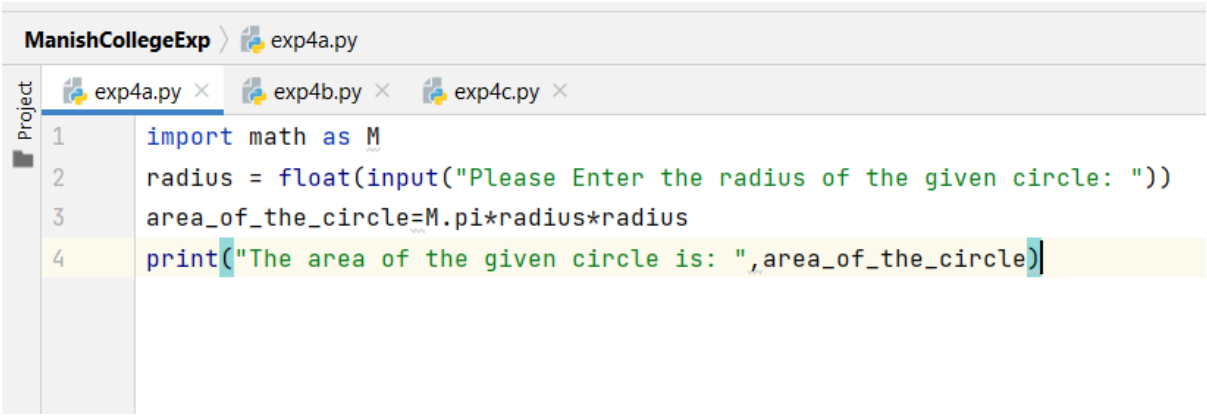


Practical Performance (4)	Writeup & Oral (4)	Attendance (2)	Total (10)

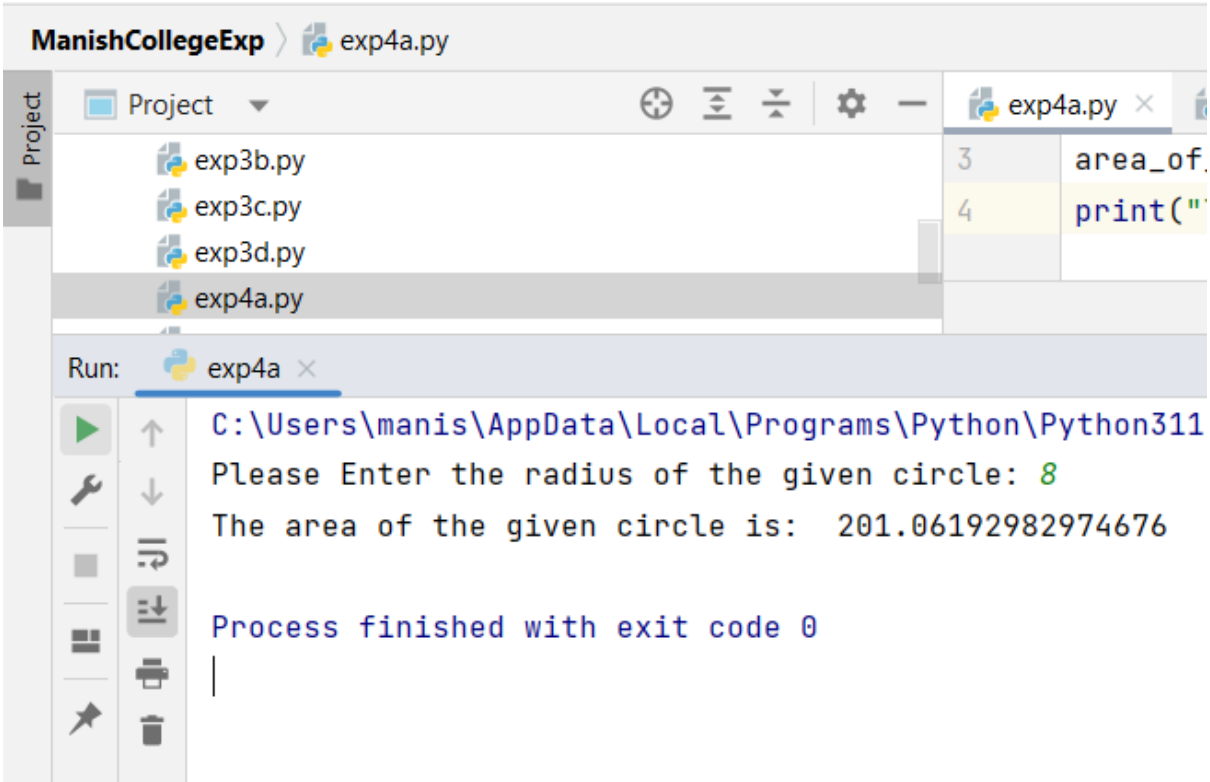
Experiment No.4

Aim: Edit/compile/run a program to read the radius of a circle and print the area of the circle.

Program A:



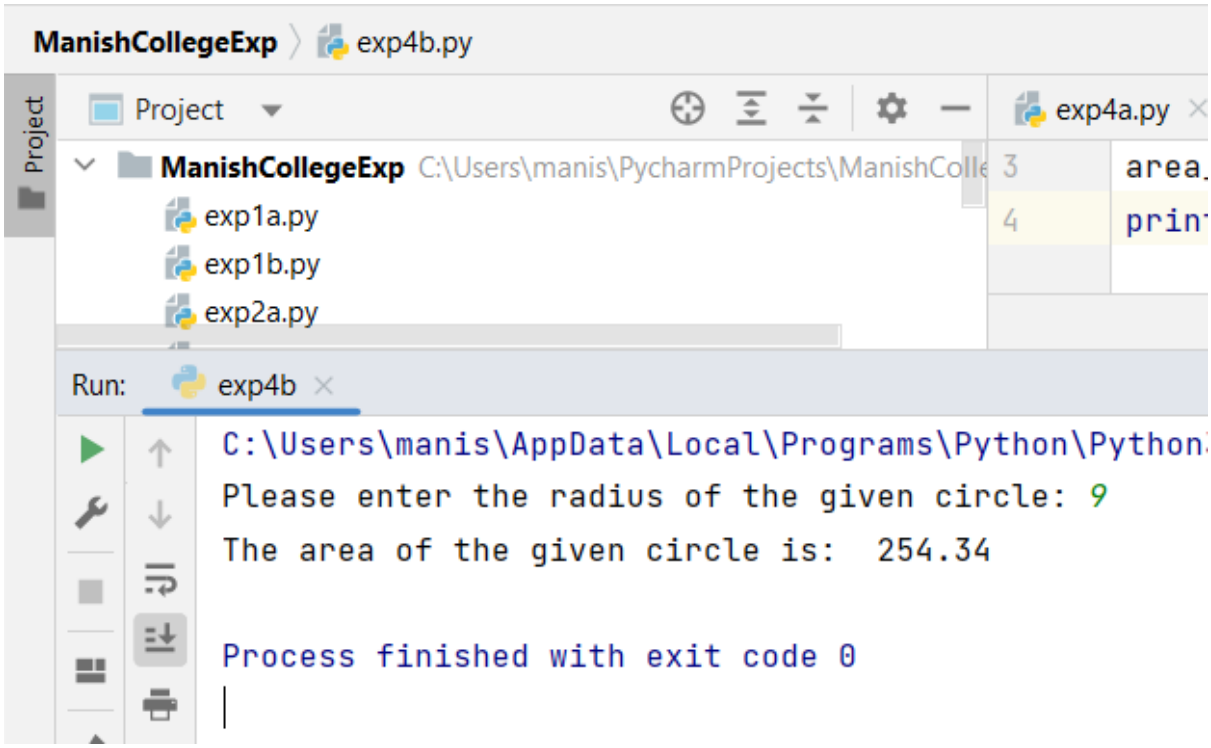
Output:



Program B:

```
pi = 3.14
Radius = float(input("Please enter the radius of the given circle: "))
area_of_the_circle = pi*Radius*Radius
print("The area of the given circle is: ",area_of_the_circle)
```

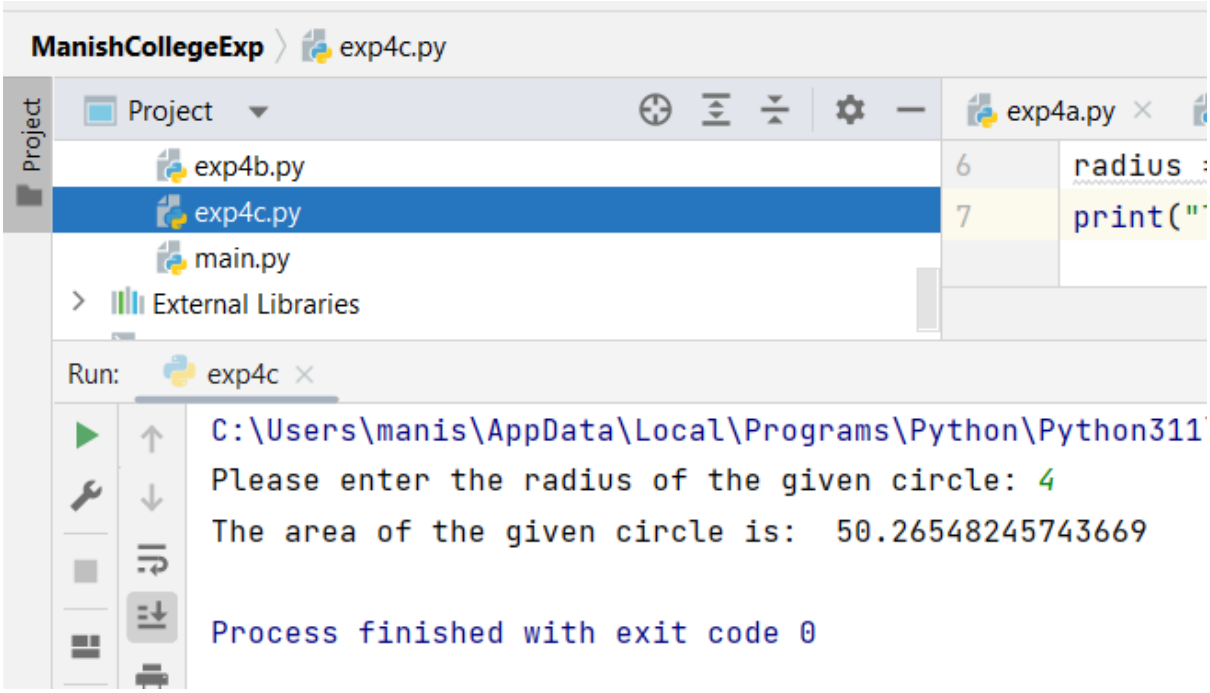
Output:



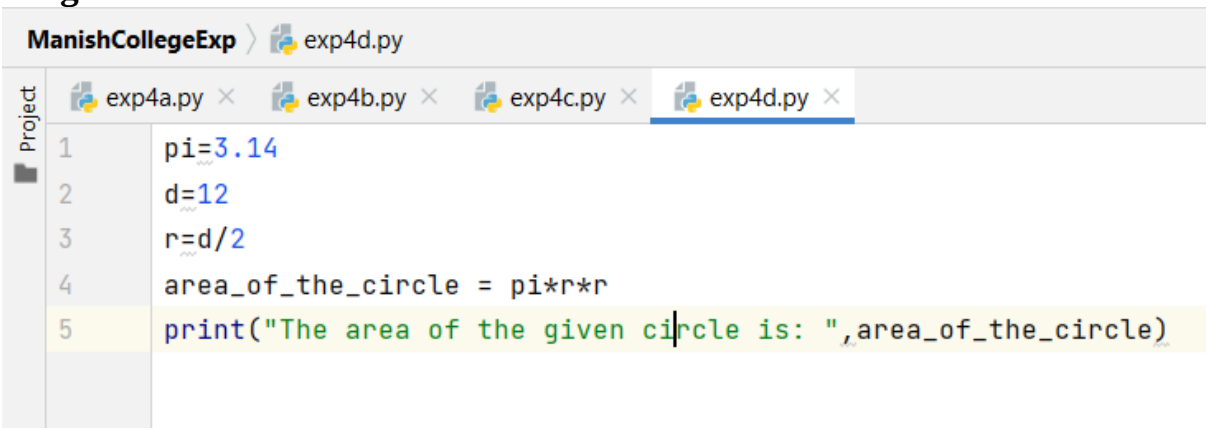
Program C:

```
import math

def area_of_the_circle (radius):
    area = radius**2*math.pi
    return area
radius = float(input("Please enter the radius of the given circle: "))
print("The area of the given circle is: ",area_of_the_circle(radius))
```

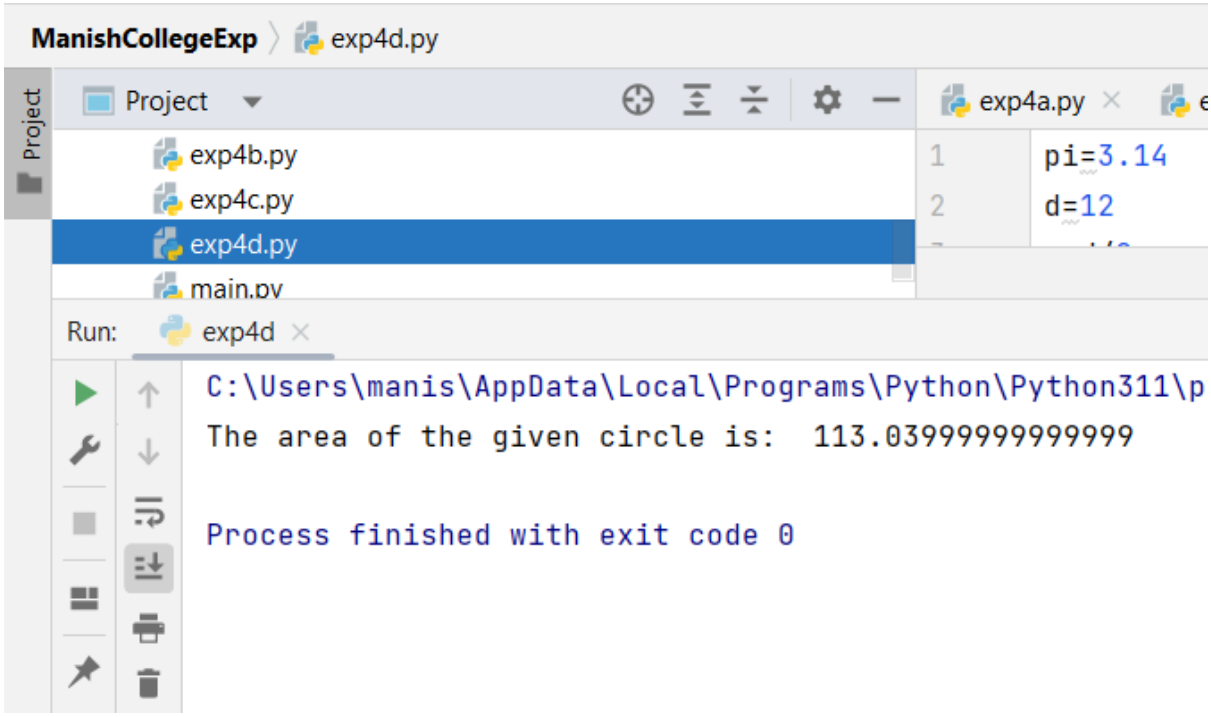
Output:



Program D:



Output:



Practical Performance (4)	Writeup & Oral (4)	Attendance (2)	Total (10)

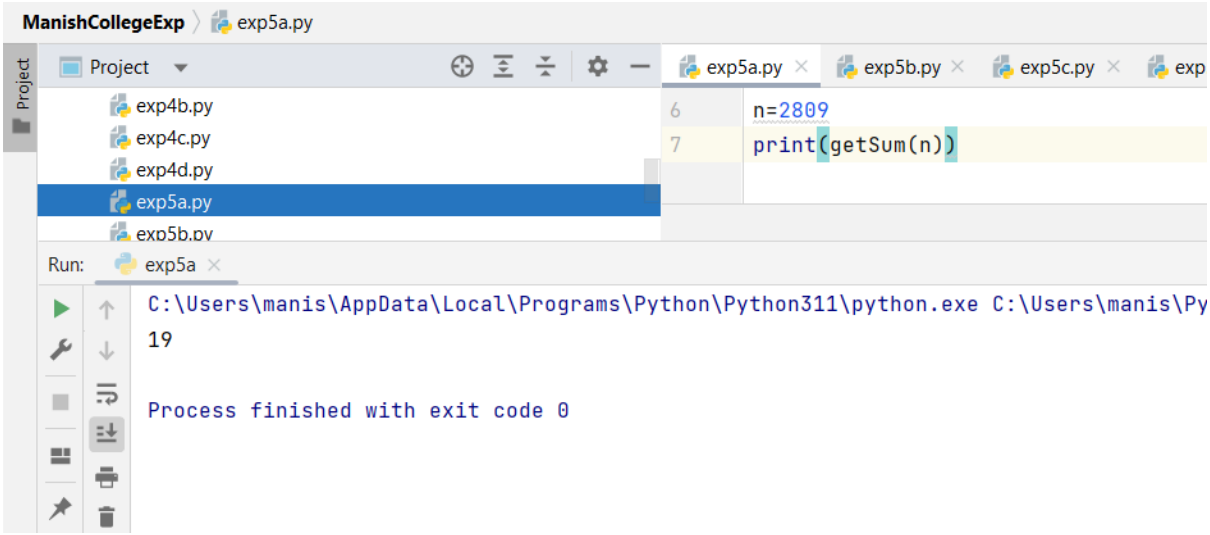
Experiment No.5

Aim: Edit/compile/run a program to read a four digit number through the keyboard and calculate the sum of its digit.

Program A:

```
def getSum(n):
    sum=0
    for digit in str(n):
        sum += int(digit)
    return sum
n=2809
print(getSum(n))
```

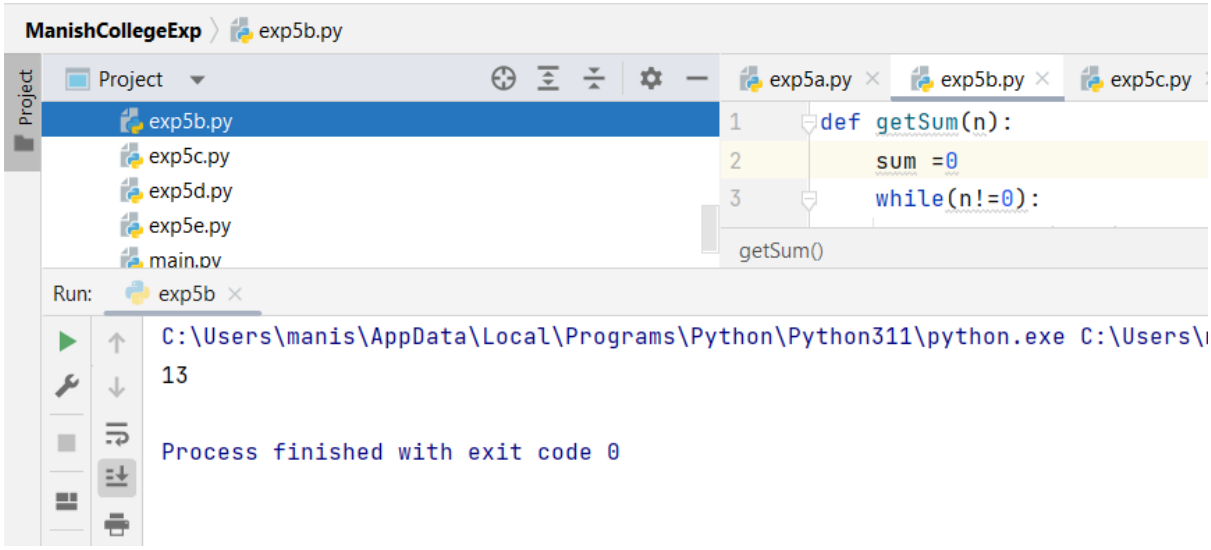
Output:



Program B:

```
def getSum(n):
    sum =0
    while(n!=0):
        sum=sum+(n%10)
        n=n//10
    return sum
n=2542
print(getSum(n))
```

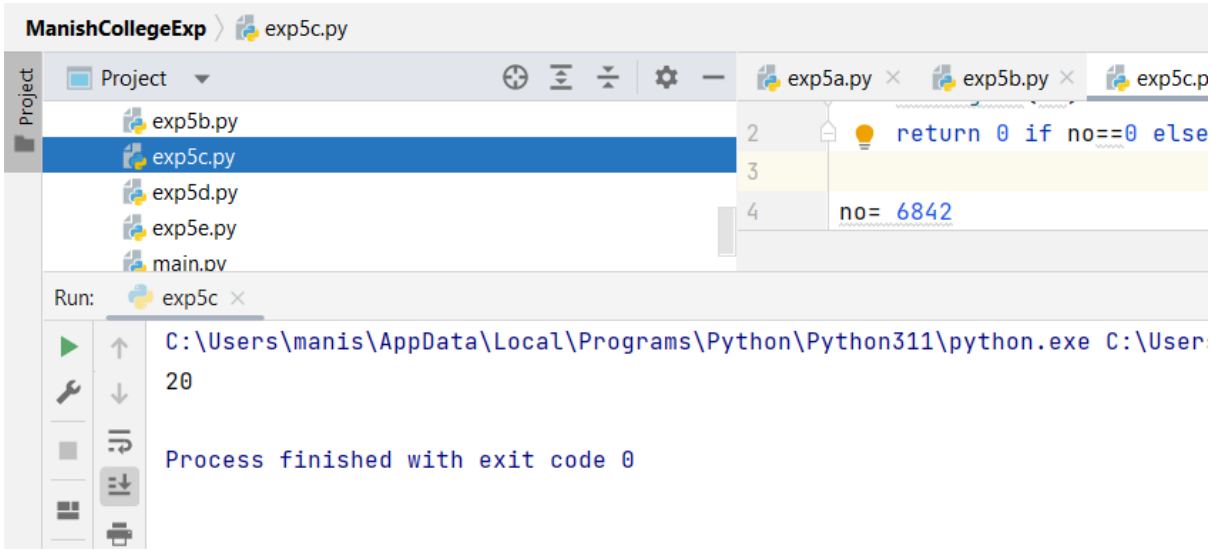
Output:



Program C:

```
def sumDigits(no):
    return 0 if no==0 else int(no%10)+sumDigits(int(no/10))

no= 6842
print(sumDigits(no))
```

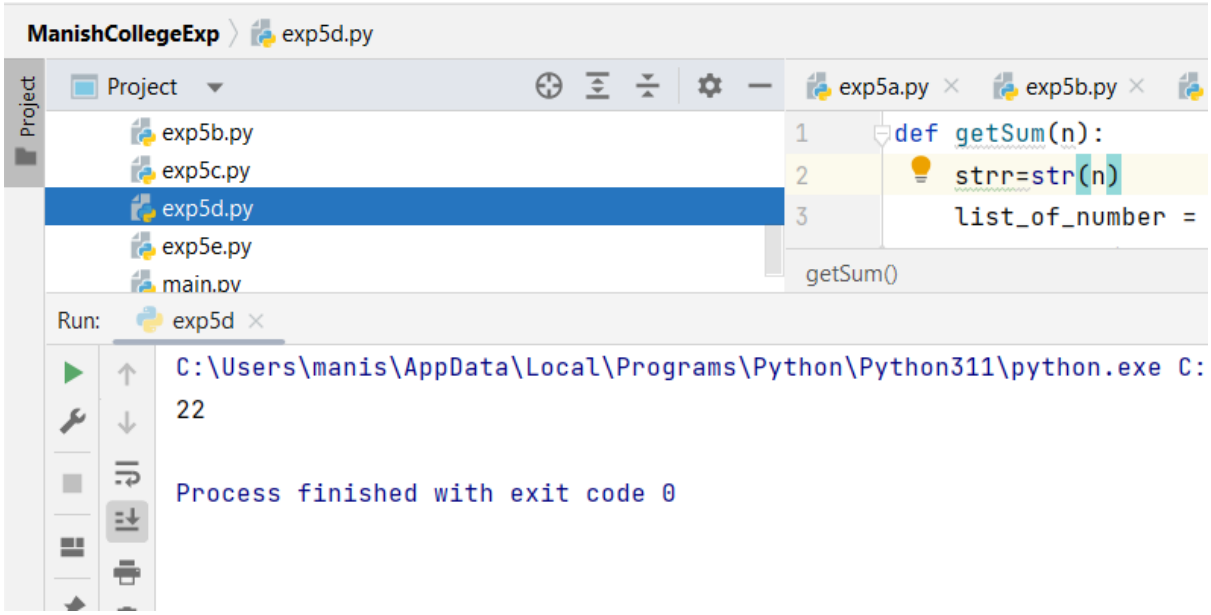
Output:



Program D:

```
def getSum(n):
    strr=str(n)
    list_of_number = list(map(int, strr.strip()))
    return sum(list_of_number)

n = 2587
print(getSum(n))
```

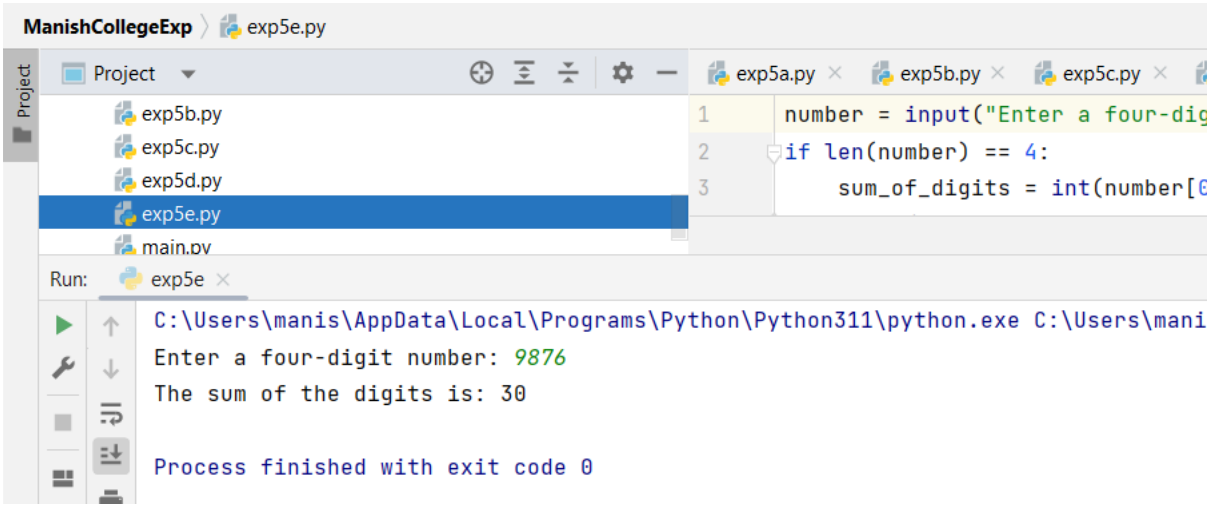
Output:



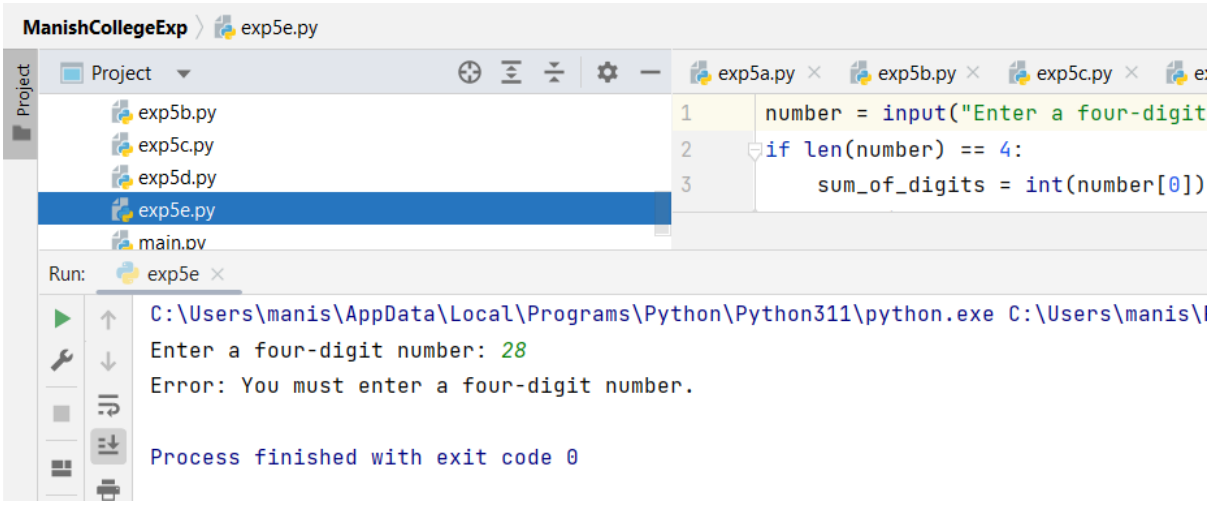
Program E:

```
number = input("Enter a four-digit number: ")
if len(number) == 4:
    sum_of_digits = int(number[0]) + int(number[1]) +
int(number[2]) + int(number[3])
    print("The sum of the digits is:", sum_of_digits)

else:
    print("Error: You must enter a four-digit number.")
```

Output: If 4 digits are entered correctly:



Output: If 4 digits are not entered:



Practical Performance (4)	Writeup & Oral (4)	Attendance (2)	Total (10)

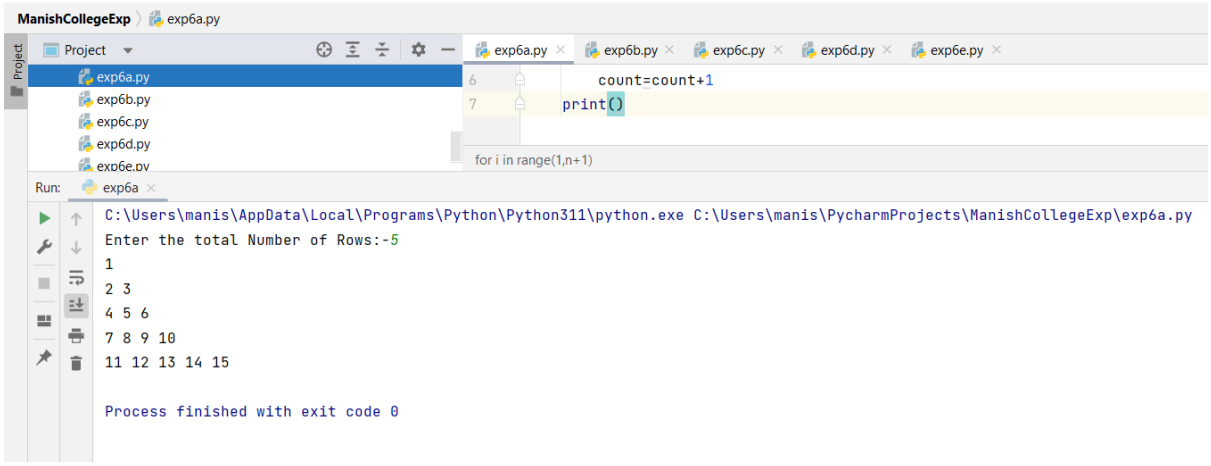
Experiment No.6

Aim: Edit/compile/run a program to display the Floyd Triangle

Program A:

```
n=int(input("Enter the total Number of Rows:-"))
count=1
for i in range(1,n+1):
    for j in range(1,i+1):
        print(count,end= ' ')
        count=count+1
    print()
```

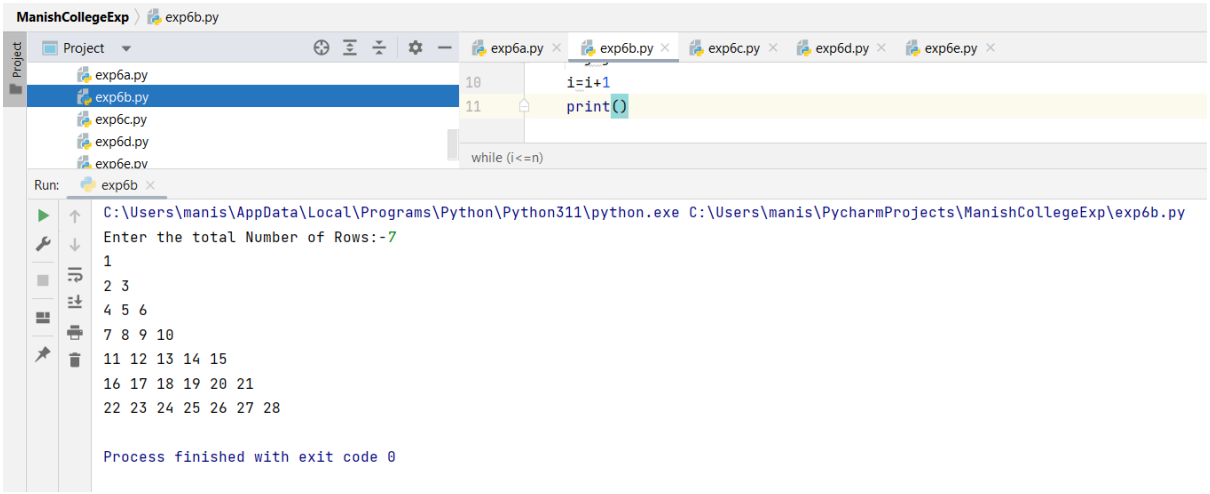
Output:



Program B:

```
n=int(input("Enter the total Number of Rows:-"))
count=1
i=1
while(i<=n):
    j=1
    while(j<=i):
      print(count,end=' ')
      count=count+1
      j=j+1
    i=i+1
    print()
```

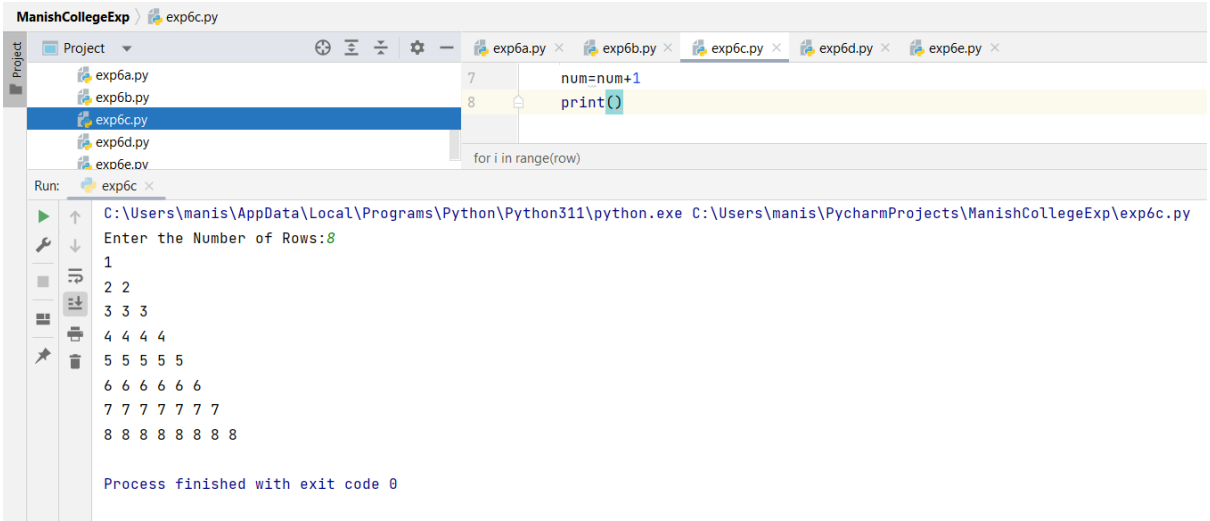
Output:



Program C:

```
print("Enter the Number of Rows:",end="")
row=int(input())
num=1
for i in range(row):
    for j in range(i+1):
        print(num,end=' ')
    num=num+1
    print()
```

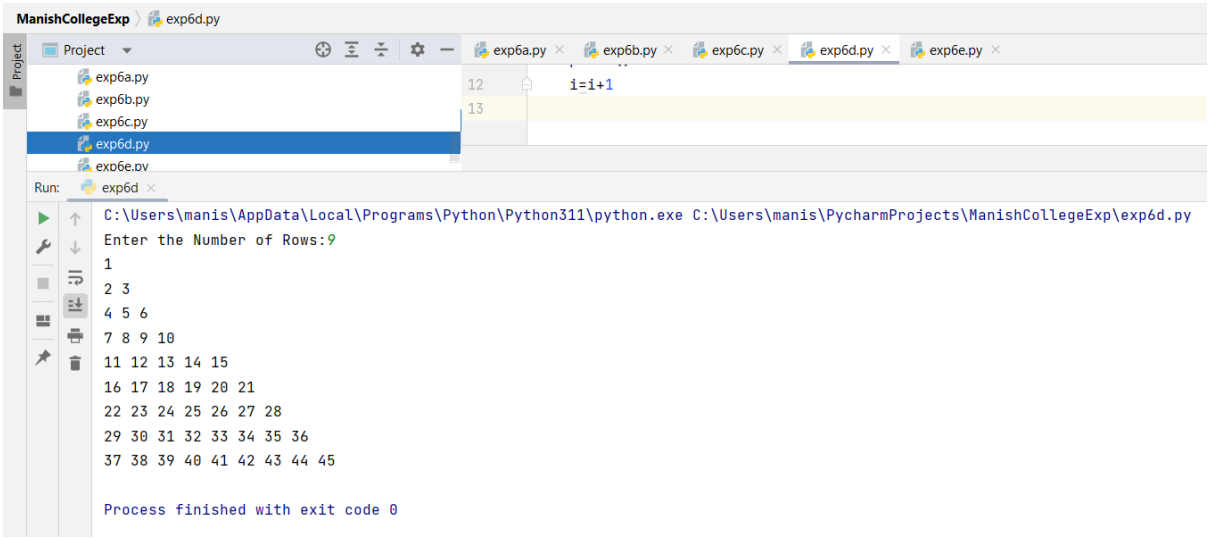
Output:



Program D:

```
print("Enter the Number of Rows:",end="")
row=int(input())
num=1
i=0
while i<row:
    j=0
    while j<i+1:
        print(num,end=' ')
        num=num+1
        j=j+1
    print()
    i=i+1
```

Output:

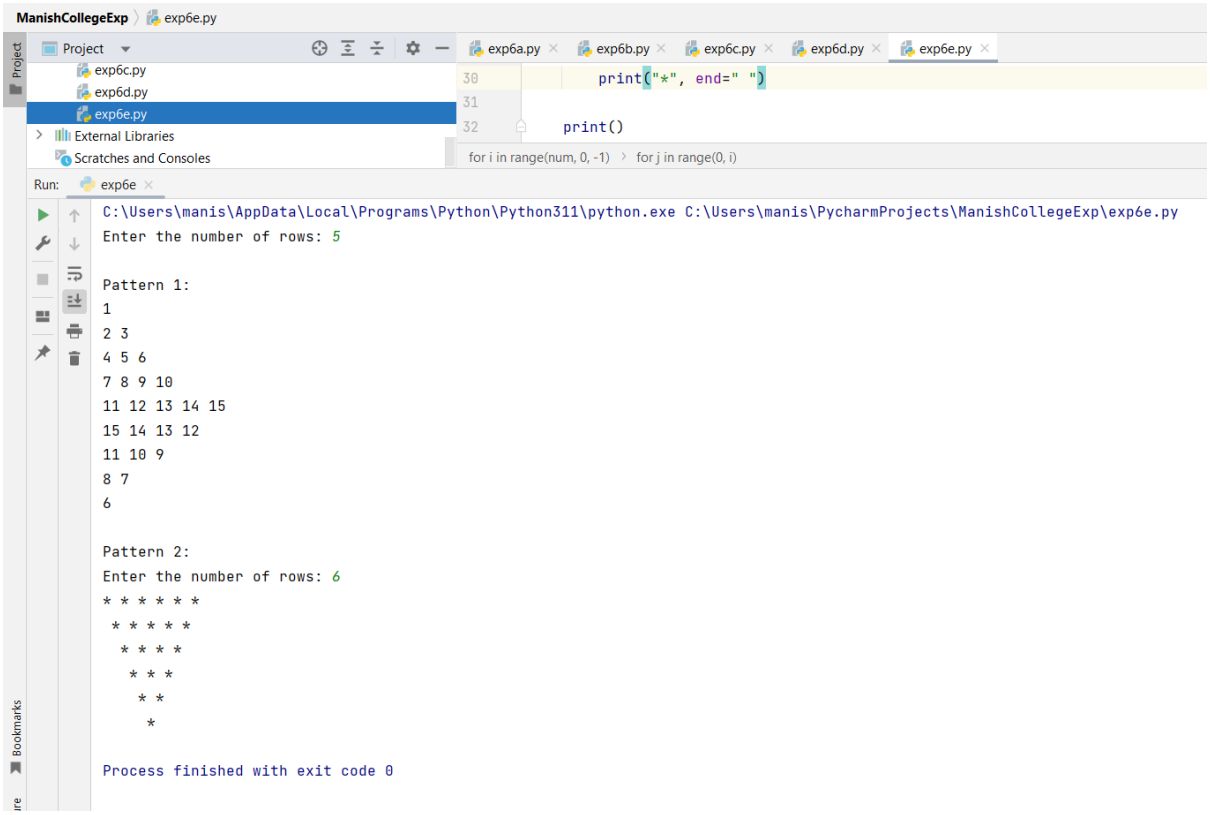


Program E:

```
rows = int(input("Enter the number of rows: "))
# Pattern 1
print("\nPattern 1:")
number = 1
for i in range(1, rows+1):
    for j in range(1, i+1):
        print(number, end=" ")
        number += 1
    print()
number -= 1
for i in range(rows-1, 0, -1):
    for j in range(i, 0, -1):
        print(number, end=" ")
        number -= 1
    print()
```

```
#Pattern 2
print("\nPattern 2:")
num = int(input("Enter the number of rows: "))
for i in range(num, 0, -1):

    for j in range(0, num - i):
        print(end=" ")

    for j in range(0, i):
        print("*", end=" ")

    print()
```

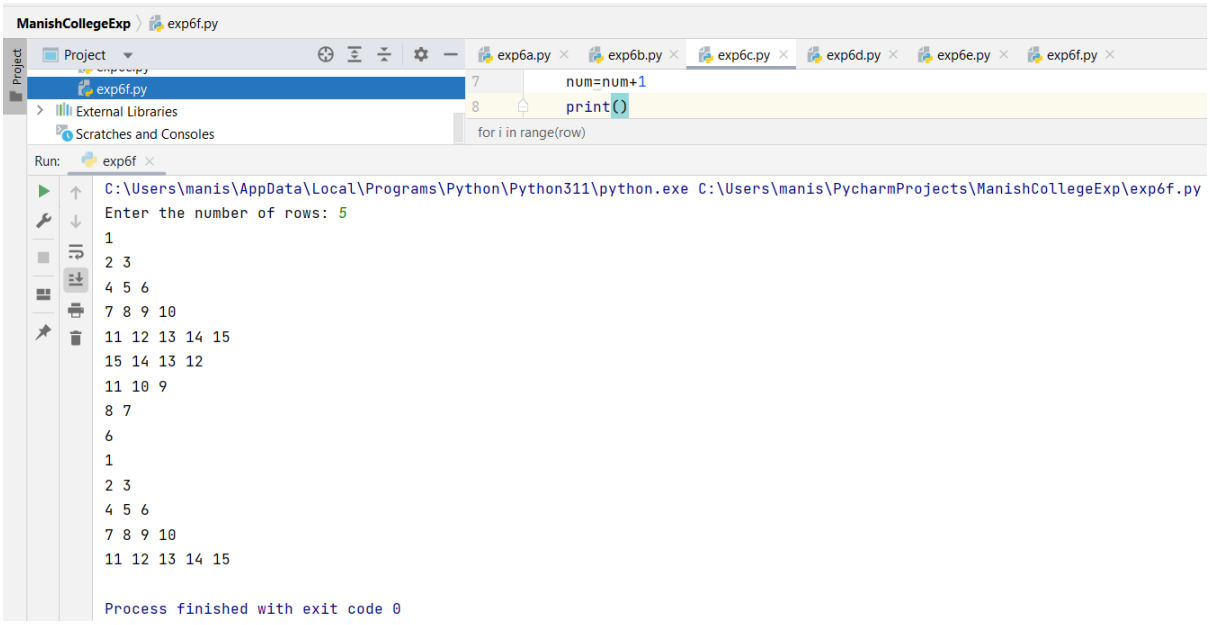
Output:



Program F:

```
row = int(input("Enter the number of rows: "))
number = 1
for i in range(1, row+1):
    for j in range(1, i+1):
        print(number, end=" ")
        number += 1
    print()
number -= 1
for i in range(row-1, 0, -1):
    for j in range(i, 0, -1):
        print(number, end=" ")
        number -= 1
    print()
number = 1
for i in range(1, row+1):
    for j in range(1, i+1):
        print(number, end=" ")
        number += 1
    print()
```

Output:



Practical Performance (4)	Writeup & Oral (4)	Attendance (2)	Total (10)

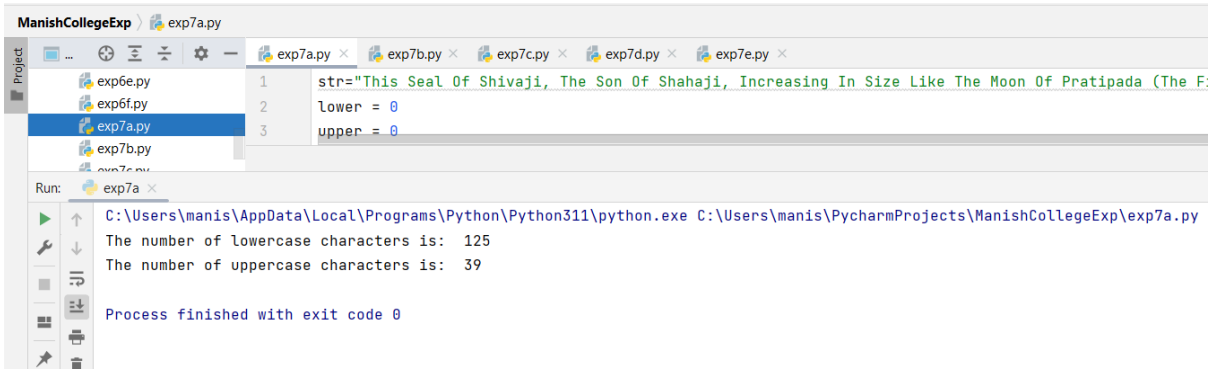
Experiment No.7

Aim: Edit/compile/run a program to read a string and display the total number of uppercase and lowercase letters.

Program A:

```
str="This Seal Of Shivaji, The Son Of Shahaji, Increasing In
Size Like The Moon Of Pratipada (The First Day After The
Moonless Night), It Will Be Worshiped By The World & It Will
Shine Only For Well Being Of People."
lower = 0
upper = 0
for i in str:
    if (i.islower()):
        lower+=1

    elif(i.isupper()):
        upper+=1
print("The number of lowercase characters is: ", lower)
print("The number of uppercase characters is: ",upper)
```

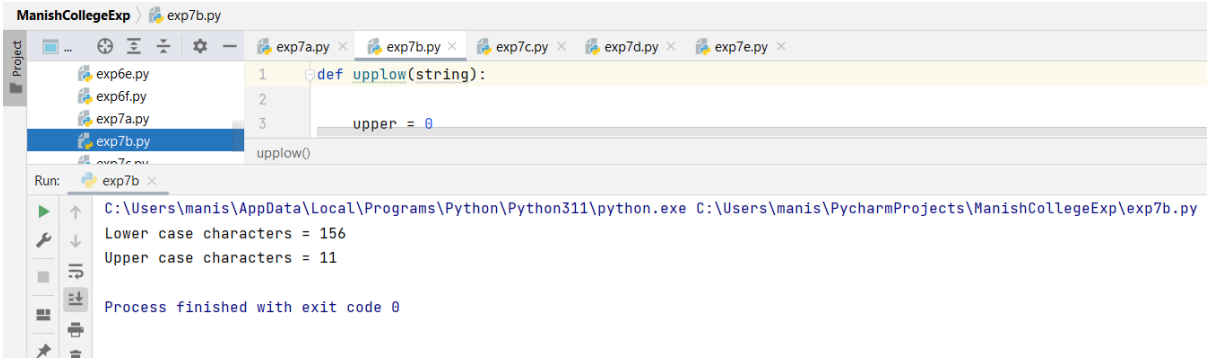
Output:



Program B:

```
def upplow(string):

    upper = 0
    lower = 0
    for i in range(len(string)):
        if(ord(string[i]) >=97 and
        ord(string[i]) <=122):
            lower+=1

        elif(ord(string[i]) >=65 and
        ord(string[i]) <=90):
            upper+=1
    print('Lower case characters = %s' %lower, "\n"
          'Upper case characters = %s' %upper)

string = "The Royal Seal of Shambho (Sambhaji), the son of
Shiv (Shivaji) is shining with glory and is limitless as the
sky. Under the shelter of this Rajmudra, everyone is
protected. Nothing is superior to this Rajmudra."
upplow(string)
```

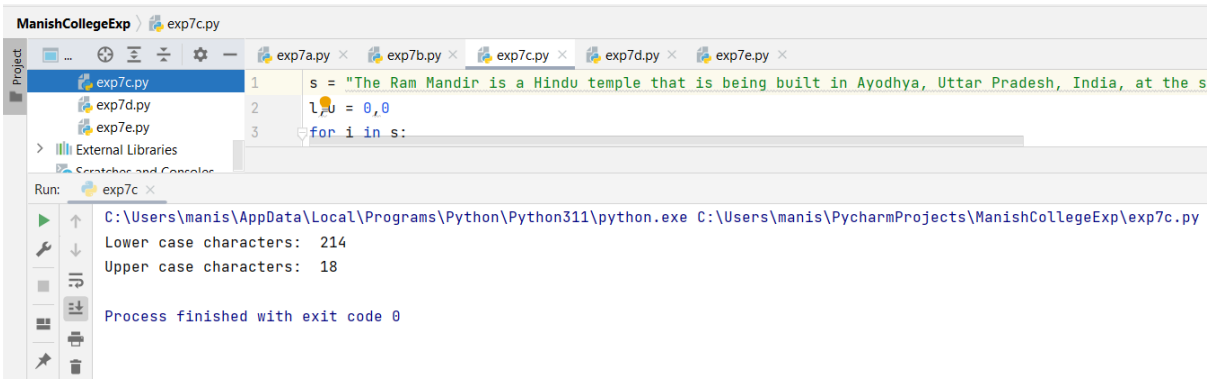
Output:



Program C:

```
s = "The Ram Mandir is a Hindu temple that is being built in
Ayodhya, Uttar Pradesh, India, at the site of Ram Janmabhoomi,
according to the Ramayana the birthplace of Rama, a principal
deity of Hinduism. The temple construction is being supervised
by the Shri Ram Janmabhoomi Teerth Kshetra."
l,u = 0,0
for i in s:
    if(i >= 'a' and i <= 'z'):
        l=l+1
    if(i>'A' and i<='Z'):
        u=u+1
print('Lower case characters: ',l)
print('Upper case characters: ',u)
```

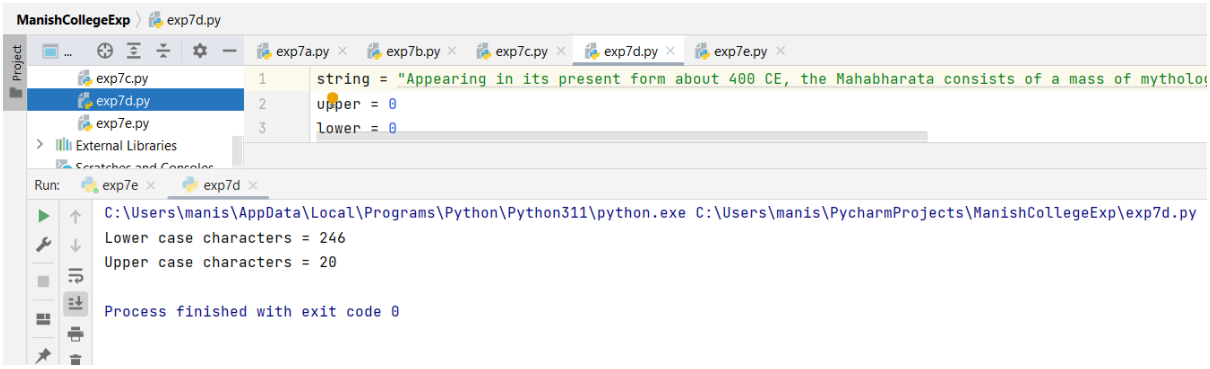
Output:



Program D:

```
string = "Appearing in its present form about 400 CE, the
Mahabharata consists of a mass of mythological and didactic
material arranged around a central heroic narrative that tells
of the Struggle for Sovereignty between Two Groups of Cousins,
The Kauravas (Sons of Dhritarashtra, The Descendant of Kuru)
and The Pandavas (Sons of Pandu)."
upper = 0
lower = 0
up = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
lo = "abcdefghijklmnopqrstuvwxyz"
for i in string:
    if i in up:
        upper+=1
    elif i in lo:
        lower+=1
print('Lower case characters = %s' %lower)
print('Upper case characters = %s' %upper)
```

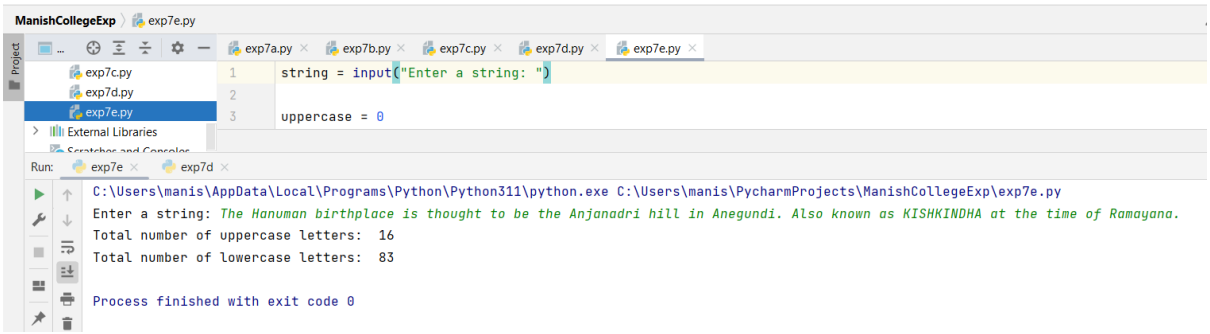
Output:



Program E:

```
string = input("Enter a string: ")
uppercase = 0
lowercase = 0
for char in string:
    if char.isupper():
        uppercase += 1
    elif char.islower():
        lowercase += 1
print("Total number of uppercase letters: ", uppercase)
print("Total number of lowercase letters: ", lowercase)
```

Output:



Practical Performance (4)	Writeup & Oral (4)	Attendance (2)	Total (10)

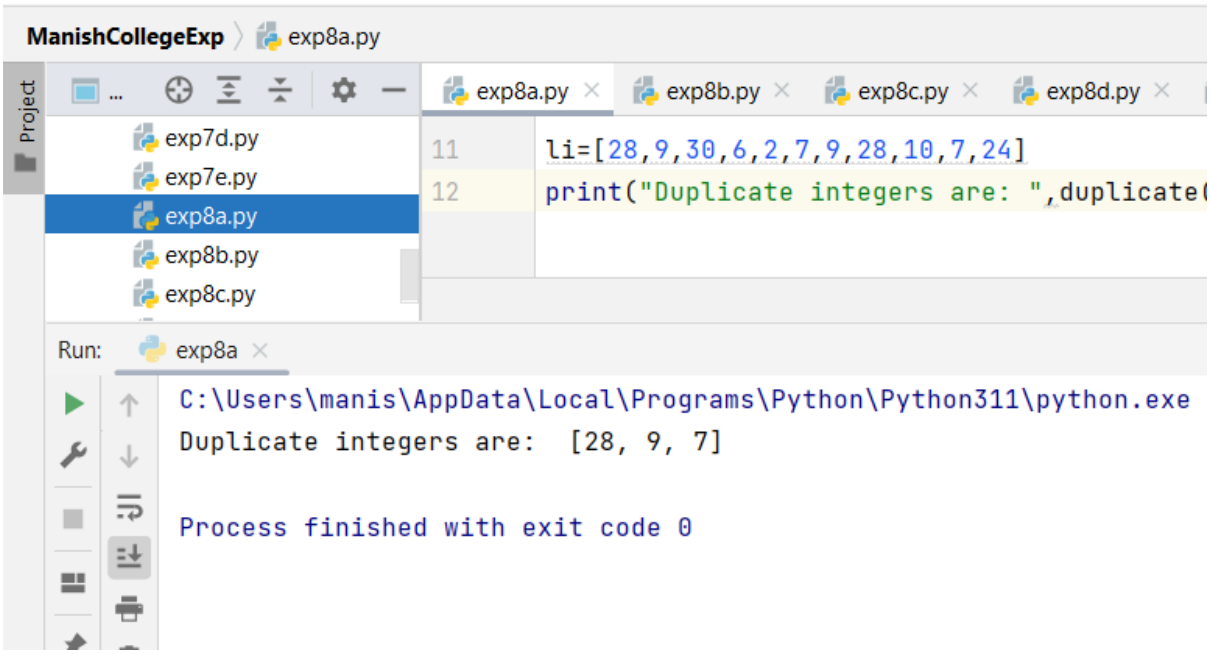
Experiment No.8

Aim: Edit/compile/run a program to duplicate all the elements of a list.

Program A: Brute force approach

```
def duplicate(li):
    n=len(li)
    dup=[]
    for i in range (n):
        k = i+1
        for j in range(k,n):
            if li[i] == li[j] and li[i] not in dup:
                dup.append(li[i])
    return dup

li=[28,9,30,6,2,7,9,28,10,7,24]
print("Duplicate integers are: ",duplicate(li))
```

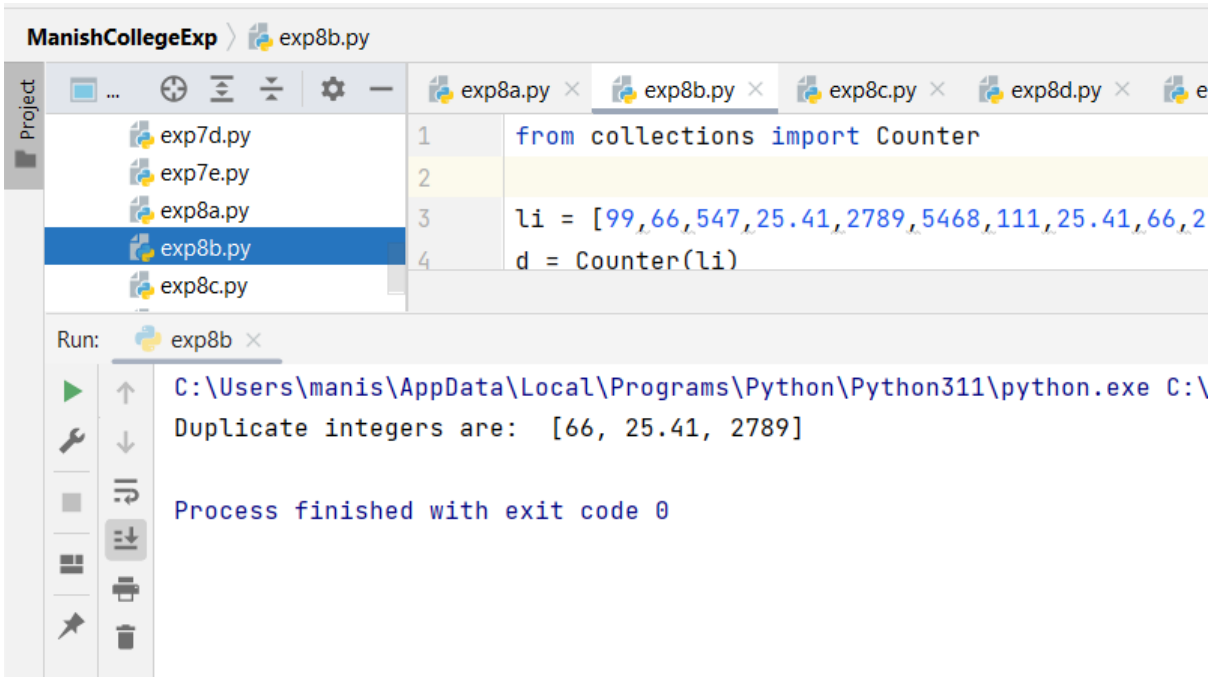
Output:



Program B: Counter function

```
from collections import Counter

li = [99,66,547,25.41,2789,5468,111,25.41,66,2789]
d = Counter(li)

repeated_list = list([num for num in d if d[num]>1])
print("Duplicate integers are: ",repeated_list)
```

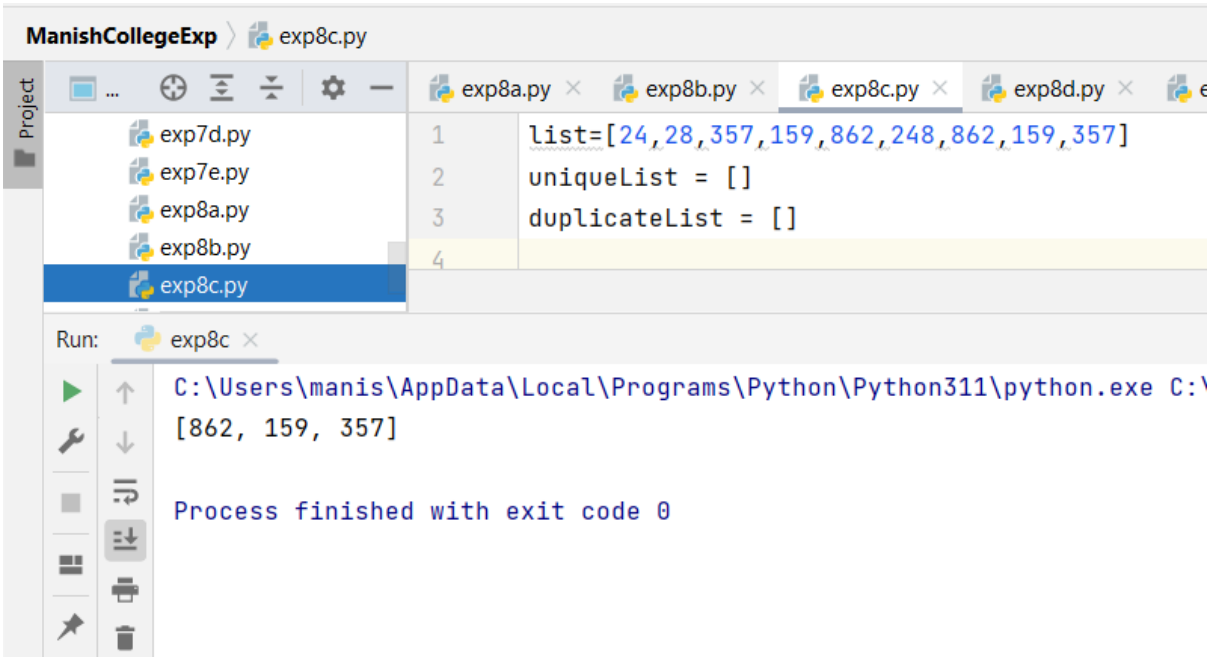
Output:



Program C: Using a single for loop

```
list=[24,28,357,159,862,248,862,159,357]
uniqueList = []
duplicateList = []

for i in list:
    if i not in uniqueList:
        uniqueList.append(i)
    elif i not in duplicateList:
        duplicateList.append(i)

print(duplicateList)
```

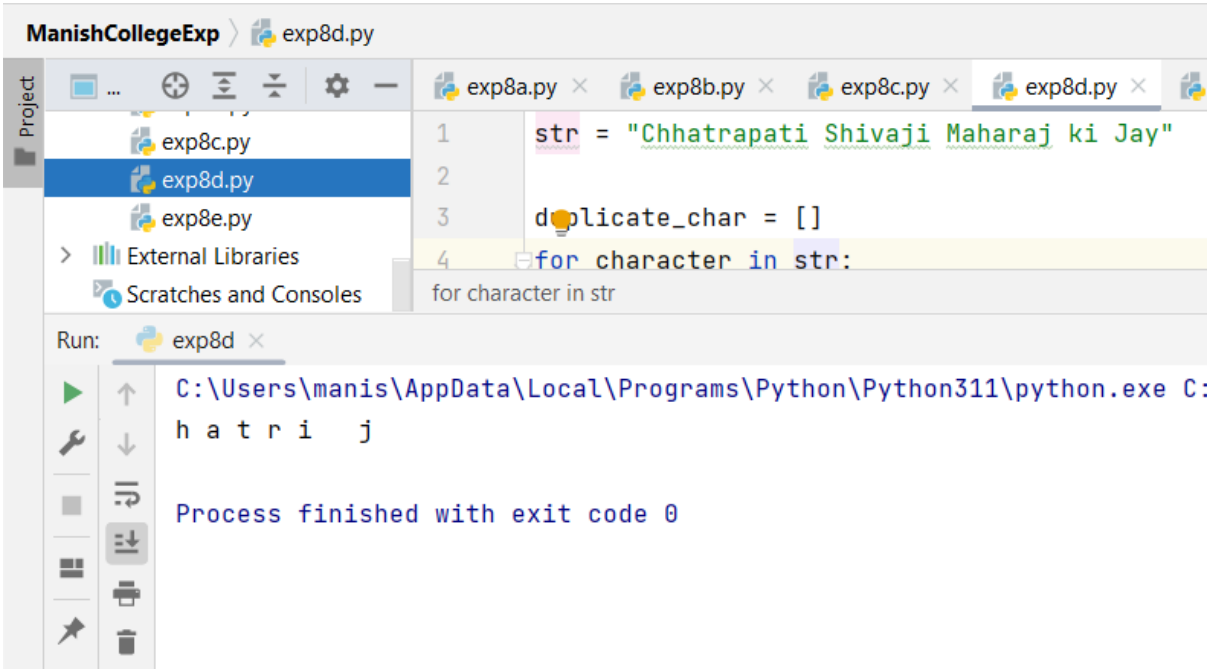
Output:



Program D: To find duplicate characters in a list

```
str = "Chhatrapati Shivaji Maharaj ki Jay"

duplicate_char = []
for character in str:

      if str.count(character) > 1:

         if character not in duplicate_char:
            duplicate_char.append(character)
print(*duplicate_char)
```

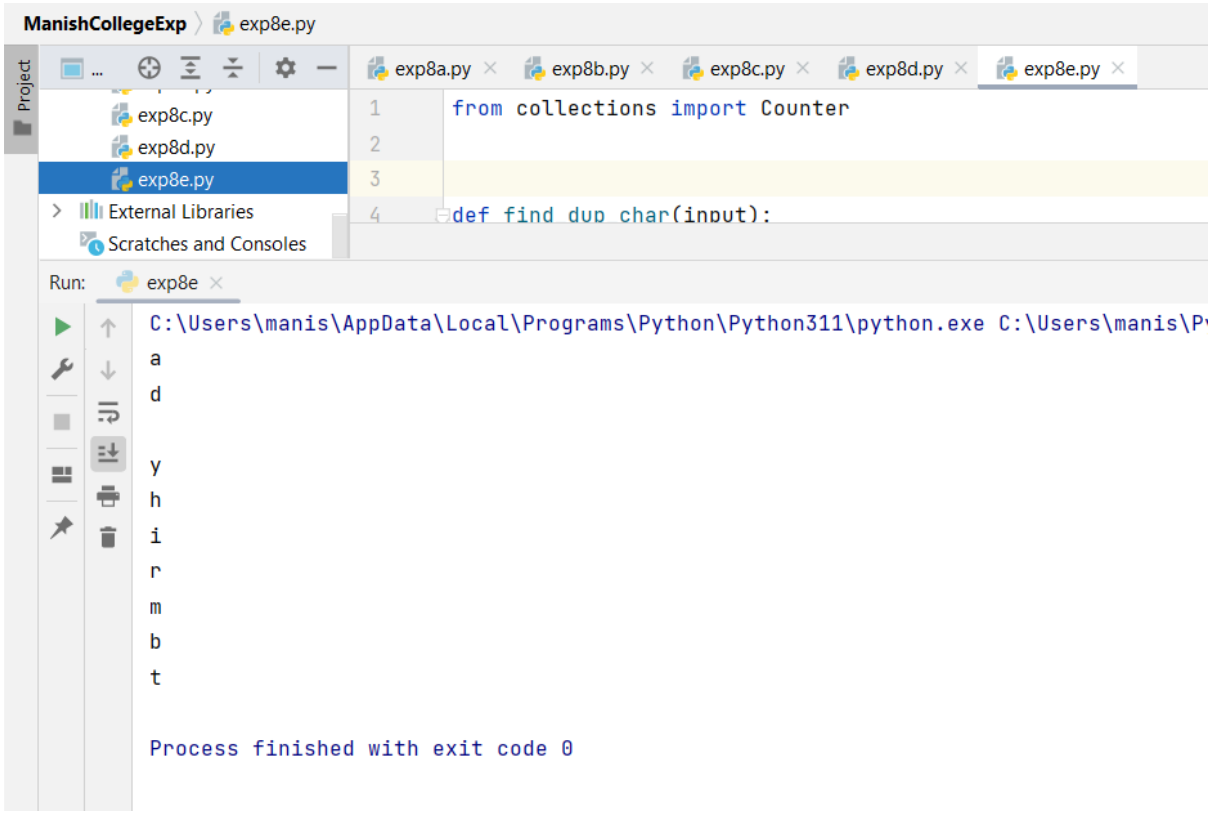
Output:



Program E: To find duplicate characters from list using Counter function.

```
from collections import Counter


def find_dup_char(input):

    WC = Counter(input)

    for letter, count in WC.items():
        if (count > 1):
            print(letter)


if __name__ == "__main__":
    input = 'Yada yada hi dharmasya glanirbhavatim bharatah'
    find_dup_char(input)
```

Output:



Practical Performance (4)	Writeup & Oral (4)	Attendance (2)	Total (10)

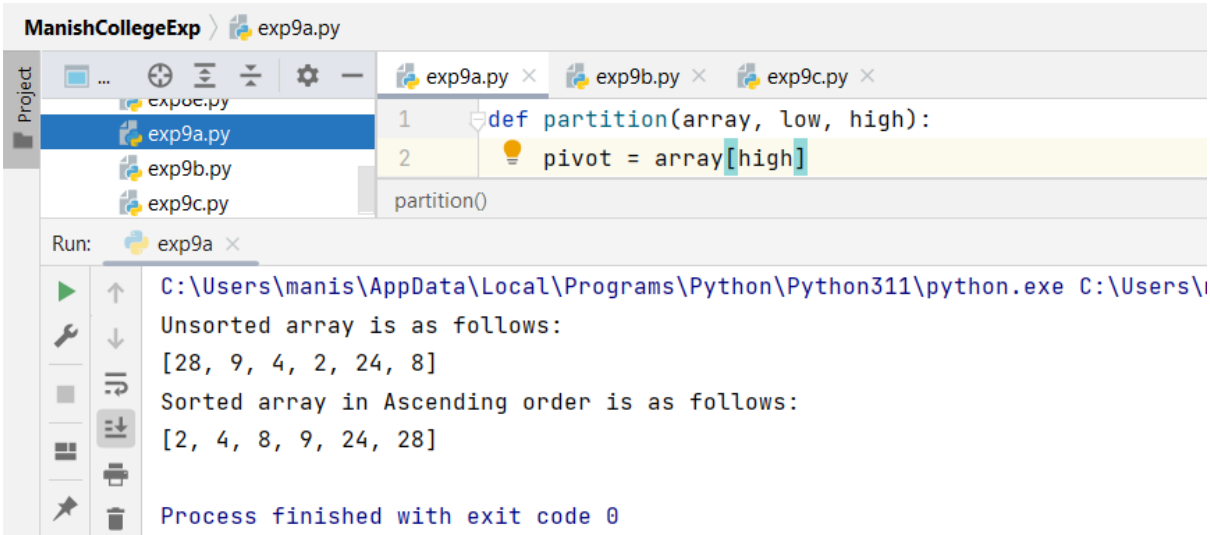
Experiment No.9

Aim: Edit/compile/run a program to implement quick sort/ merge sort/ bubble sort.

Program A: QUICK SORT ALGORITHM:

```
def partition(array, low, high):
    pivot = array[high]
    i = low - 1
    for j in range(low, high):
        if array[j] <= pivot:
            i = i + 1
            (array[i], array[j]) = (array[j], array[i])
    (array[i+1], array[high]) = (array[high], array[i+1])
    return i+1

def quickSort(array, low, high):
    if low < high:
        pi = partition(array, low, high)
        quickSort(array, low, pi-1)
        quickSort(array, pi+1, high)

data = [28,9,4,2,24,8]
print("Unsorted array is as follows: ")
print(data)
size = len(data)
quickSort(data, 0, size-1)
print('Sorted array in Ascending order is as follows: ')
print(data)
```

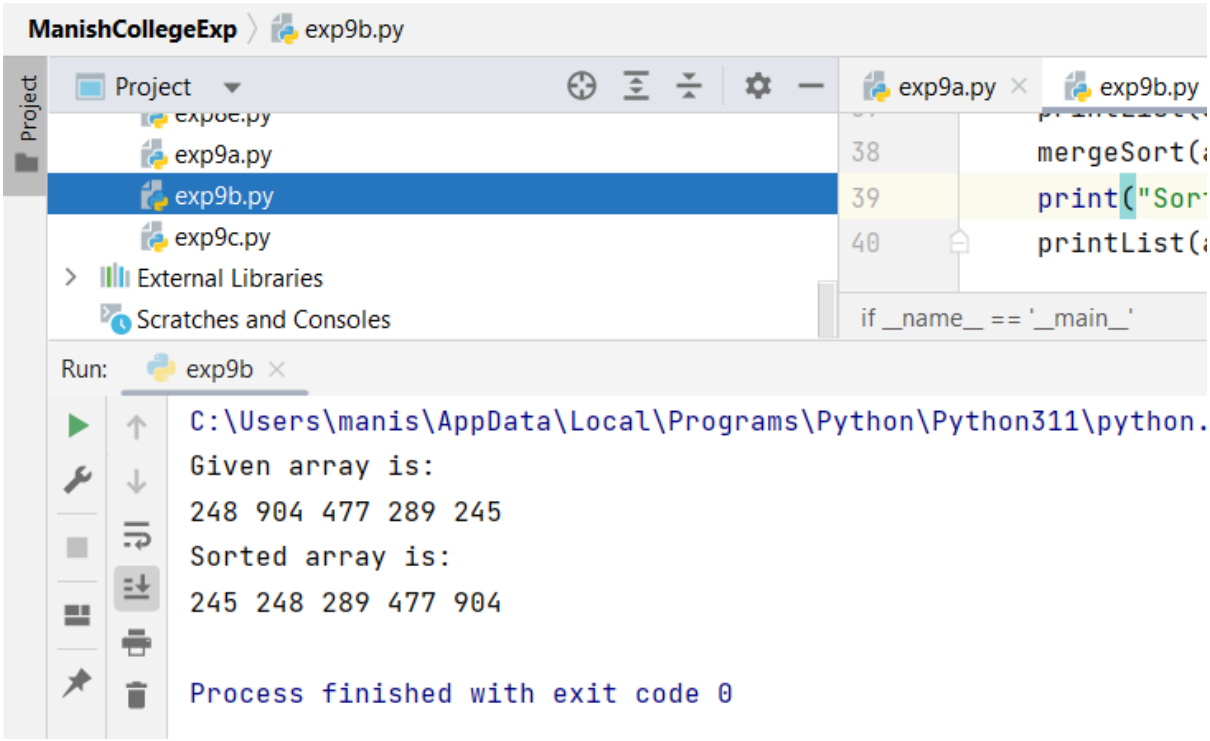
Output:



Program B: MERGE SORT ALGORITHM

```
def mergeSort(arr):
    if len(arr) > 1:
        mid = len(arr)//2
        L=arr[:mid]
        R=arr[mid:]
        mergeSort(L)
        mergeSort(R)
        i=j=k=0
        while i<len(L) and j<len(R):
            if L[i] <= R[j]:
                arr[k] = L[i]
                i+=1
            else:
                arr[k] = R[j]
                j+=1
            k+=1

        while i<len(L):
            arr[k] = L[i]
            i+=1
            k+=1

        while j<len(R):
            arr[k] = R[j]
            j+=1
            k+=1


def printList(arr):
    for i in range(len(arr)):
        print(arr[i], end=" ")
    print()

if __name__ == '__main__':
    arr=[248,904,477,289,245,]
    print("Given array is: ", end="\n")
    printList(arr)
    mergeSort(arr)
    print("Sorted array is: ", end="\n")
    printList(arr)
```

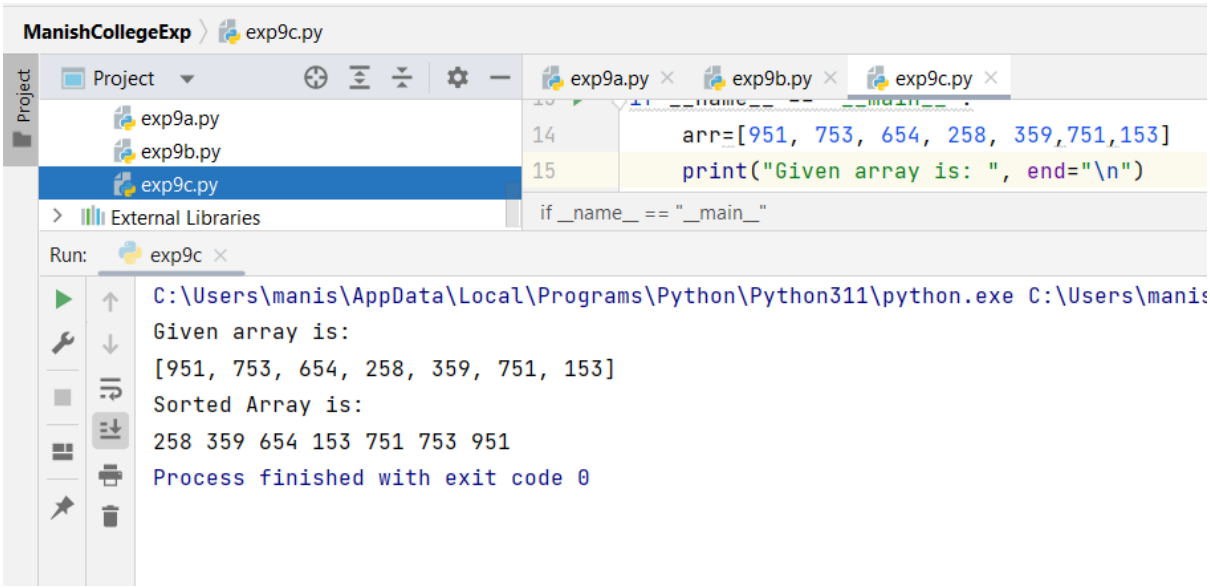
Output:



Program C: BUBBLE SORT ALGORITHM

```
def bubbleSort(arr):
    n =len(arr)
    for i in range(n):
        swapped = False

        for j in range(0,n-i-1):
            if arr[j]>arr[j+1]:
                arr[j], arr[j+1] = arr[j+1], arr[j]
                swapped = True
            if(swapped == False):
                break

if __name__ == "__main__":
    arr=[951, 753, 654, 258, 359,751,153]
    print("Given array is: ", end="\n")
    print(arr)

    bubbleSort(arr)

    print("Sorted Array is: ")
    for i in range(len(arr)):
        print("%d" %arr[i], end=" ")
```

Output:



Practical Performance (4)	Writeup & Oral (4)	Attendance (2)	Total (10)

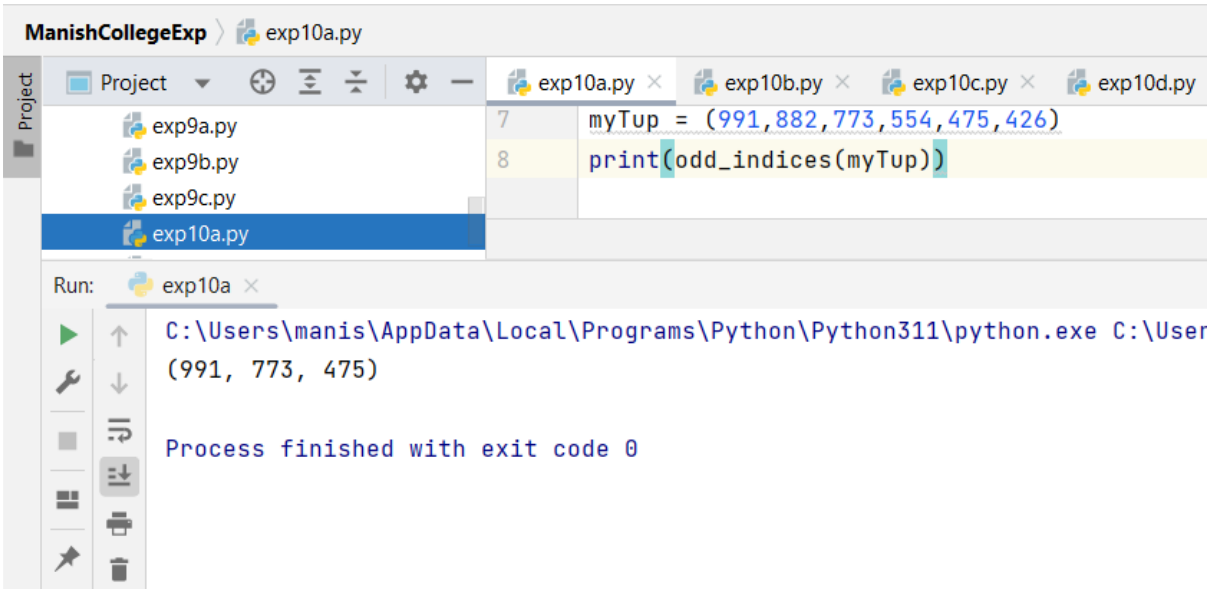
Experiment No.10

Aim: Write a function which takes a tuple as a parameter and returns a new tuple as the output, where every other element of the input tuple is copied, starting from the first one.

Program A:

```
def odd_indices(tup):
    Odd_Tup = ()
    for i in range(len(tup)):
        if i % 2 != 1:
            Odd_Tup += (tup[i],)
    return Odd_Tup
myTup = (991,882,773,554,475,426)
print(odd_indices(myTup))
```

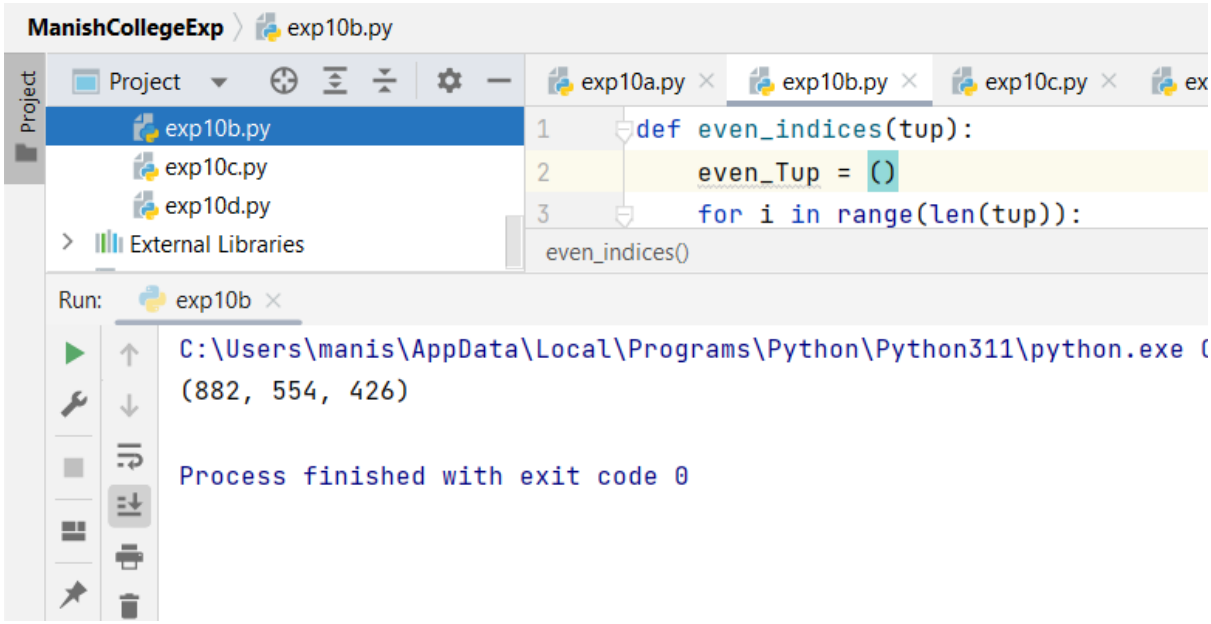
Output:



Program B:

```
def even_indices(tup):
    even_Tup = ()
    for i in range(len(tup)):
        if i % 2 != 0:
            even_Tup += (tup[i],)
    return even_Tup
myTup = (991,882,773,554,475,426)
print(even_indices(myTup))
```

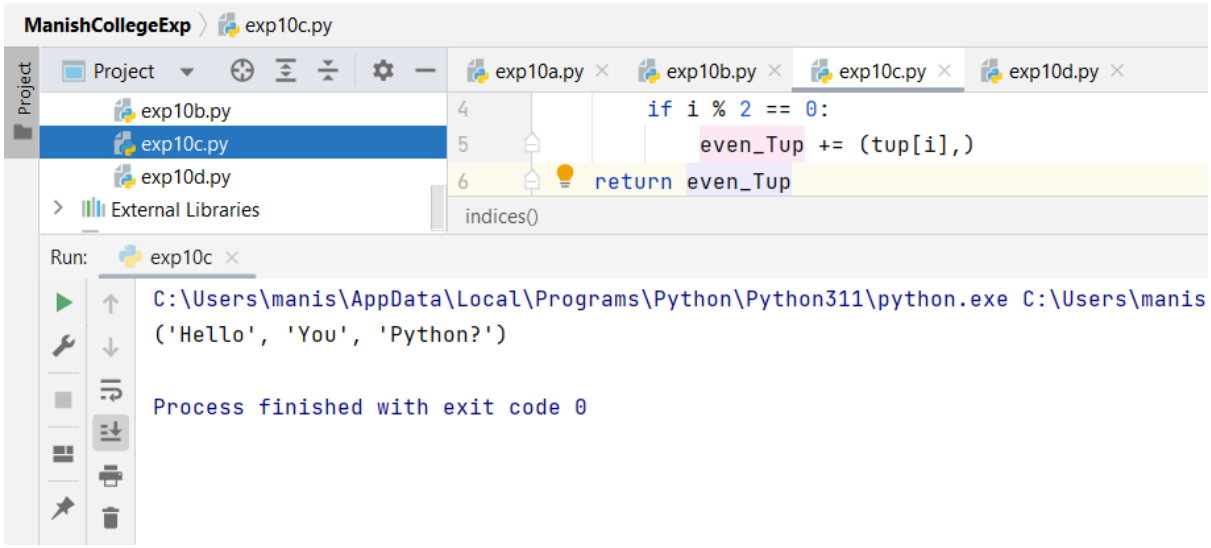
Output:



Program C: Exercise:

```
def indices(tup):
    even_Tup = ()
    for i in range(len(tup)):
        if i % 2 == 0:
            even_Tup += (tup[i],)
    return even_Tup
myTup = ('Hello','Are','You','Loving','Python?')
print(indices(myTup))
```

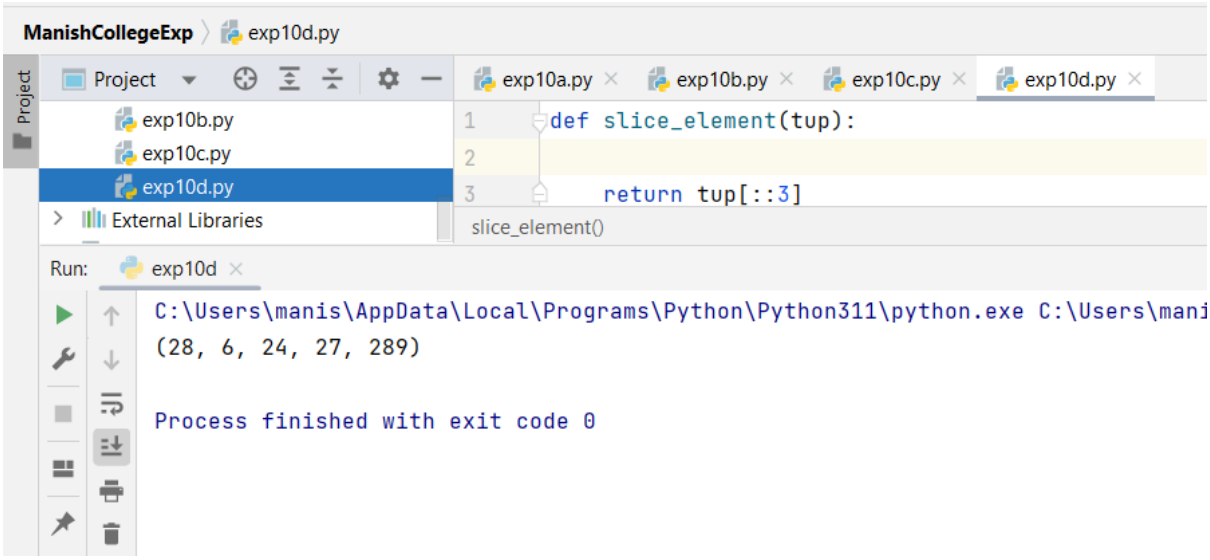
Output:



Program D: Accessing Values in Tuples using Splice Operation:

```
def slice_element(tup):

    return tup[::3]

input_tuple = (28,9,30,6,2,7,24,8,10,27,3,16,289)
output_tuple = slice_element(input_tuple)
print(output_tuple)
```

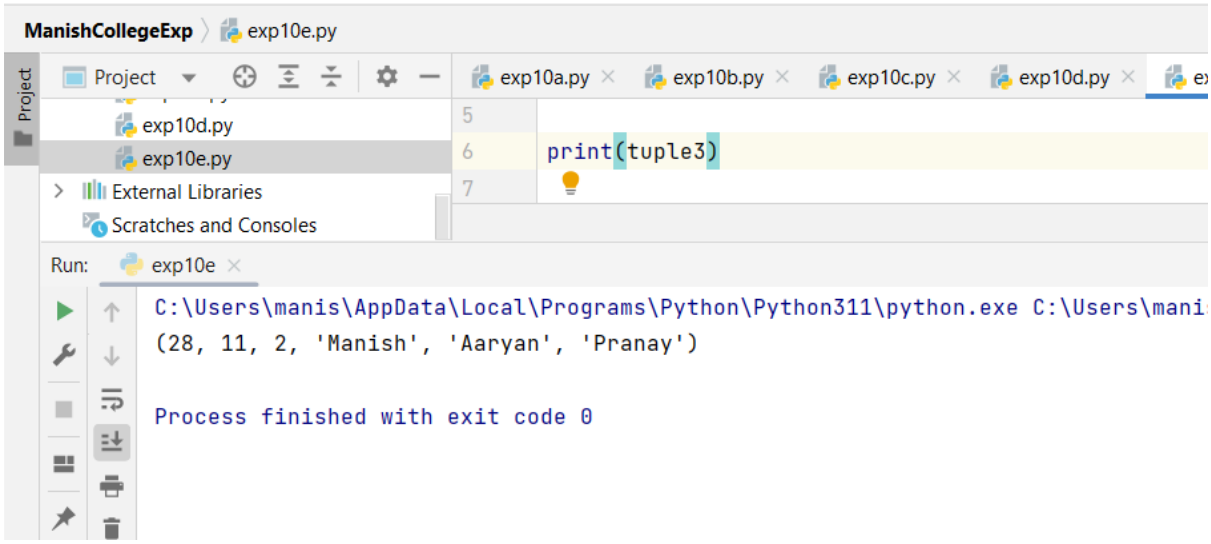
Output:



Program E: Concatenating Tuples:

```
tuple1 = (28, 11, 2)
tuple2 = ('Manish', 'Aaryan', 'Pranay')

tuple3 = tuple1 + tuple2

print(tuple3)
```

Output:



Practical Performance (4)	Writeup & Oral (4)	Attendance (2)	Total (10)

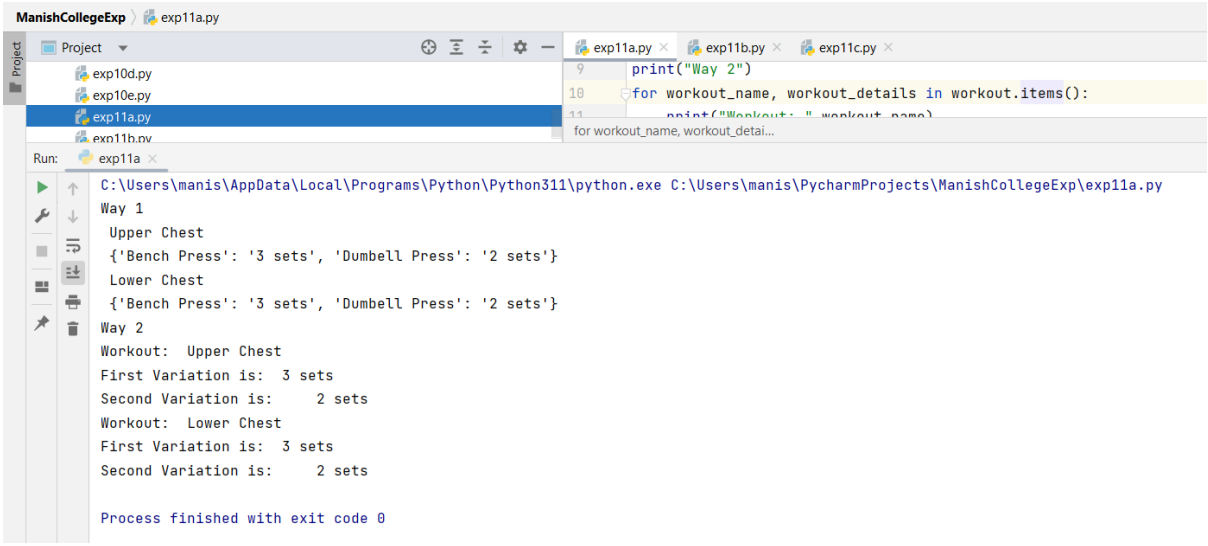
Experiment No.11

Aim: Write a function which takes a tuple as a parameter and returns a new tuple as the output, where every other element of the input tuple is copied, starting from the first one.

Program A:

```
workout = {"Upper Chest": {"Bench Press": "3 sets", "Dumbell
Press":"2 sets"},
           "Lower Chest":{"Bench Press": "3 sets", "Dumbell
Press":"2 sets"}}

print("Way 1")
for workout_name, workout_details in workout.items():
    print("", workout_name)
    print("", workout_details)

print("Way 2")
for workout_name, workout_details in workout.items():
    print("Workout: ",workout_name)
    print("First Variation is:\t", workout_details["Bench
Press"])
    print("Second Variation is:\t", workout_details["Dumbell
Press"])
```

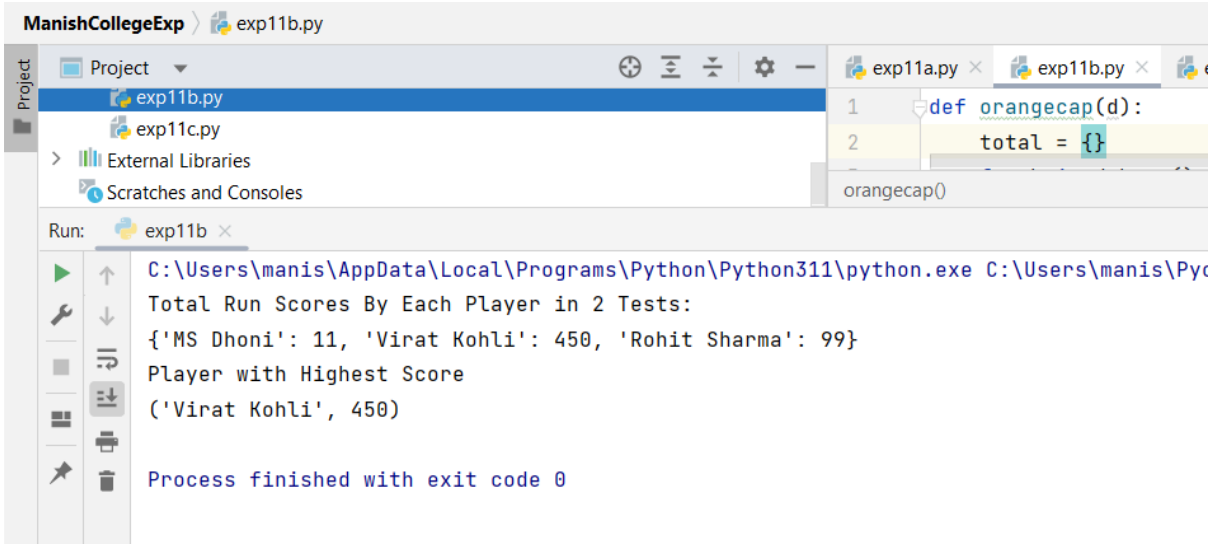
Output:



Program B:

```
def orangecap(d):
    total = {}
    for k in d.keys():
        for n in d[k].keys():
            if n in total.keys():
                total[n] = total[n] + d[k][n]
            else:
                total[n] = d[k][n]
    print('Total Run Scores By Each Player in 2 Tests: ')
    print(total)

    print('Player with Highest Score')
    maxtotal = -1
    for n in total.keys():
        if total[n]>maxtotal:
            maxname = n
            maxtotal = total[n]
    return (maxname, maxtotal)

d=orangecap({'Test 1':{'MS Dhoni': 10,'Virat Kohli':300},
'Test 2':{'MS Dhoni':1,'Virat Kohli':150,'Rohit Sharma':99}})
print(d)
```

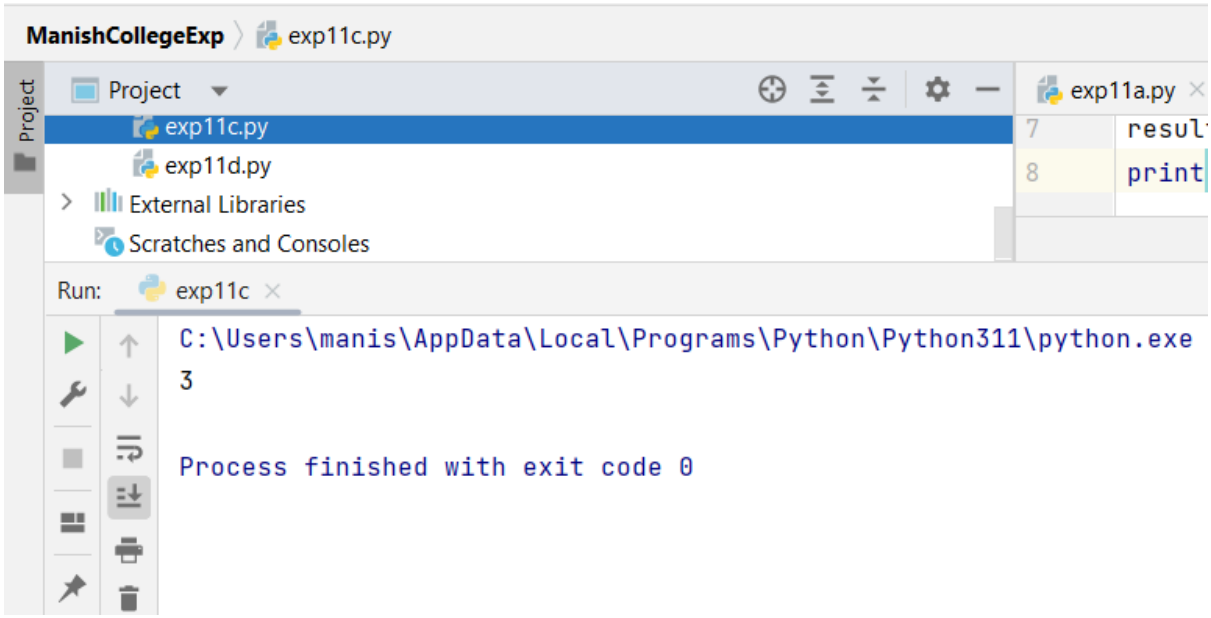
Output:



Program C: Exercise:

```
def how_many(animals):
    count = 0
    for i in animals:
        count += len(animals[i])
    return count
animals = {'L':['Lion'],'D':['Donkey'],'E':['Elephant']}
result = how_many(animals)
print(result)
```

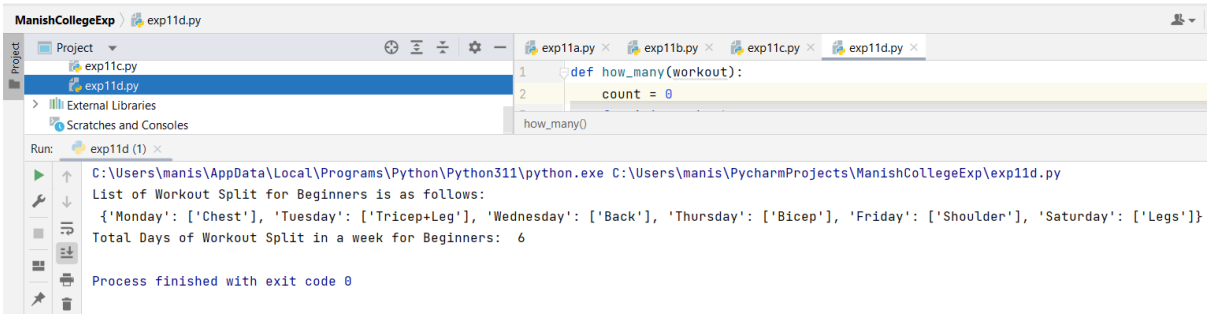
Output:



Program D:

```
def how_many(workout):
    count = 0
    for i in workout:
        count += len(workout[i])
    return count
workout =
{'Monday':['Chest'],'Tuesday':['Tricep+Leg'],'Wednesday':['Bac
k'],'Thursday':['Bicep'],'Friday':['Shoulder'],'Saturday':['Le
gs']}
print("List of Workout Split for Beginners is as follows:
\n",workout)
result = how_many(workout)
print("Total Days of Workout Split in a week for Beginners:
",result)
```

Output:



Practical Performance (4)	Writeup & Oral (4)	Attendance (2)	Total (10)