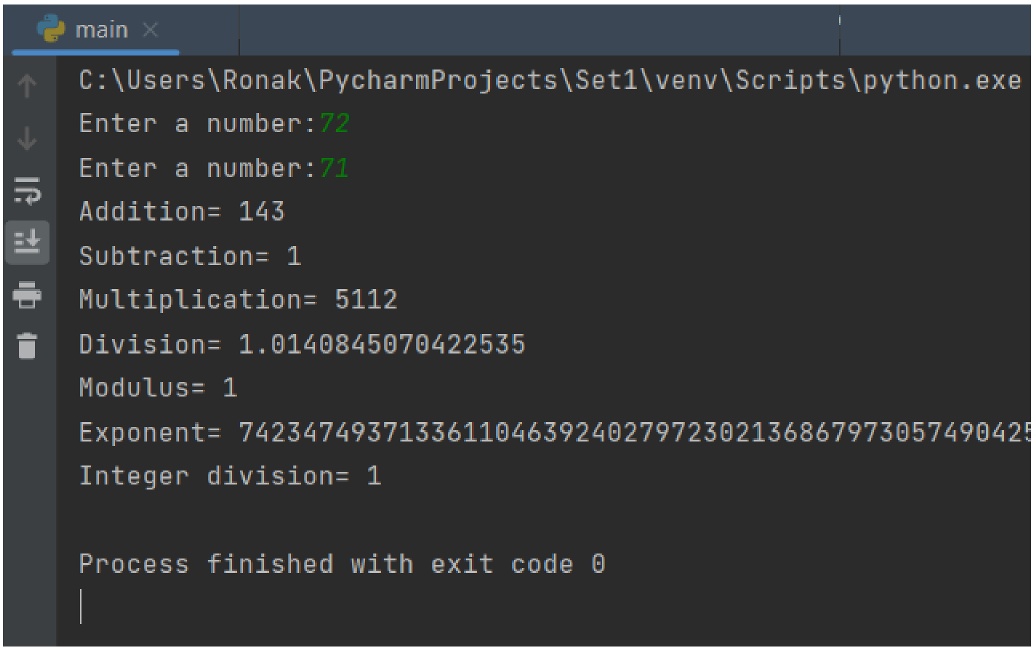
Practical Set-1: Basics of Python

Practical -1

Aim: Write a python program to create a simple arithmetic application including operations(addition, subtraction, multiplication, division, modulus, exponent, integer division).

```
a=int(input("Enter a number:"))

b=int(input("Enter a number:"))

print("Addition=",(a+b))

print("Subtraction=",(a-b))

print("Multiplication=",(a*b))

print("Division=",(a/b))

print("Modulus=",(a%b))

print("Exponent=",(a**b))

print("Integer division=",(a//b))
```

Output:



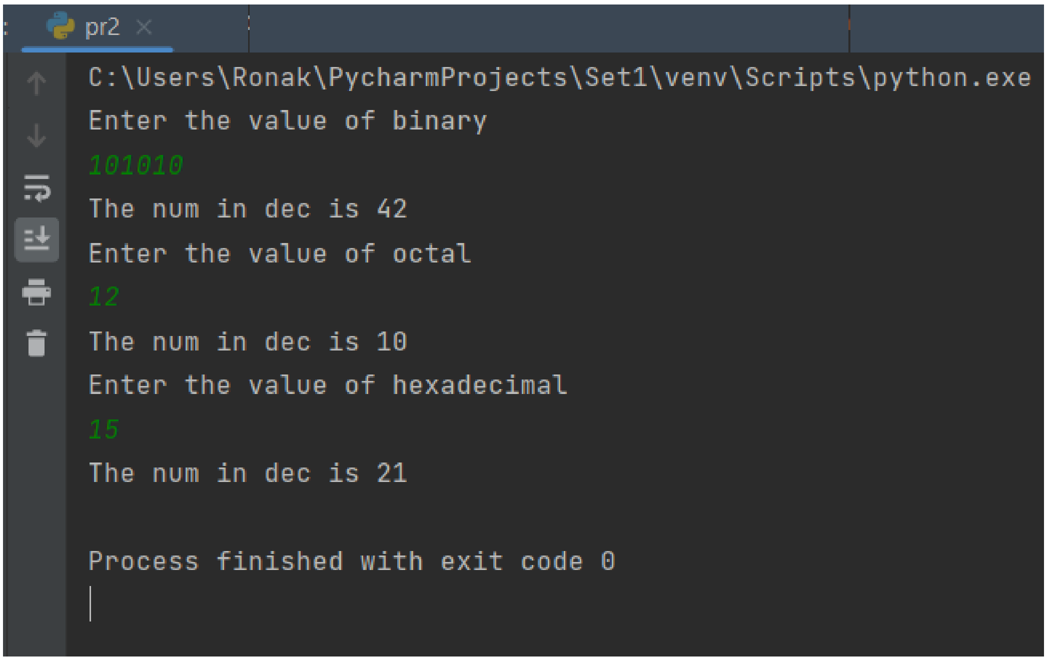
```
C:\Users\Ronak\PycharmProjects\Set1\venv\Scripts\python.exe
Enter a number:72
Enter a number:71
Addition= 143
Subtraction= 1
Multiplication= 5112
Division= 1.0140845070422535
Modulus= 1
Exponent= 7423474937133611046392402797230213686797305749042
Integer division= 1

Process finished with exit code 0
```

Practical -2

Aim: Write a python program to convert numbers from octal, binary and hexadecimal systems into decimal number system.

```
a=input("Enter the value of binary \n")
print("The num in dec is",int(a,2))
a=input("Enter the value of octal \n")
print("The num in dec is",int(a,8))
a=input("Enter the value of hexadecimal \n")
print("The num in dec is",int(a,16))
```

Output:



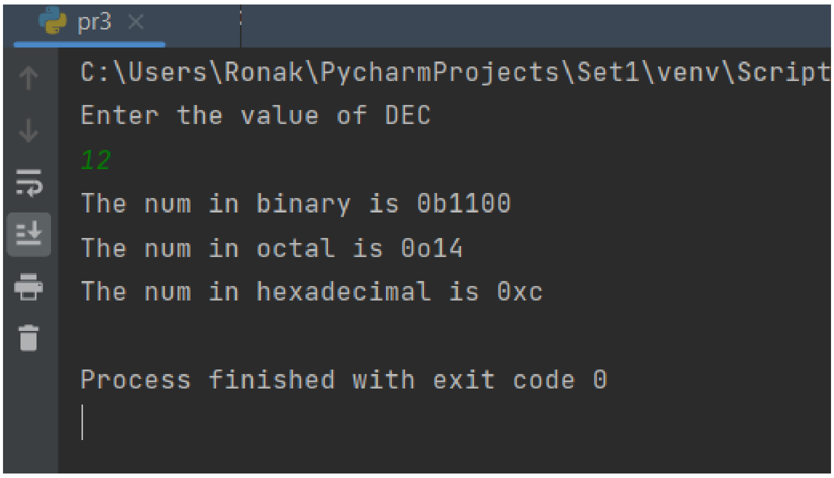
```
C:\Users\Ronak\PycharmProjects\Set1\venv\Scripts\python.exe
Enter the value of binary
101010
The num in dec is 42
Enter the value of octal
12
The num in dec is 10
Enter the value of hexadecimal
15
The num in dec is 21

Process finished with exit code 0
```

Practical -3

Aim: Write a python program to convert numbers from decimal number system into octal, binary and hexadecimal system.

```
a=int(input("Enter the value of DEC \n"))

print("The num in binary is",bin(a))

print("The num in octal is",oct(a))

print("The num in hexadecimal is",hex(a))
```

Output:



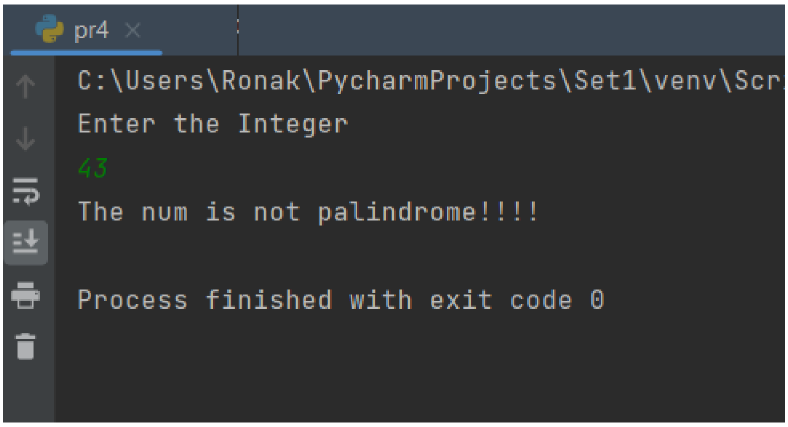
```
C:\Users\Ronak\PycharmProjects\Set1\venv\Script
Enter the value of DEC
12
The num in binary is 0b1100
The num in octal is 0o14
The num in hexadecimal is 0xc

Process finished with exit code 0
```

Practical -4

Aim: Write a python program to check whether the given number is palindrome or not.

```
no=int(input("Enter the Integer \n"))
sum=0
temp=no
while(no>0):
    r=no%10
    sum=(sum*10)+r
    no=int(no/10)
if(temp==sum):
    print(f"The num is paalindrome {no}")
else:
    print("The num is not palindrome!!!!")
```

Output:



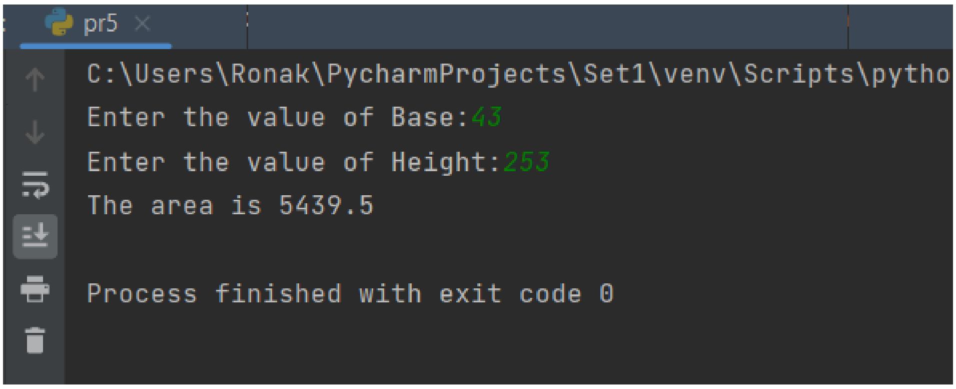
```
C:\Users\Ronak\PycharmProjects\Set1\venv\Scr
Enter the Integer
43
The num is not palindrome!!!!

Process finished with exit code 0
```

Practical -5

Aim: Write a python program to calculate area of a triangle.

```
x=float(input("Enter the value of Base:"))

y=float(input("Enter the value of Height:"))

Area=(x*y)/2

print("The area is",Area)
```

Output:



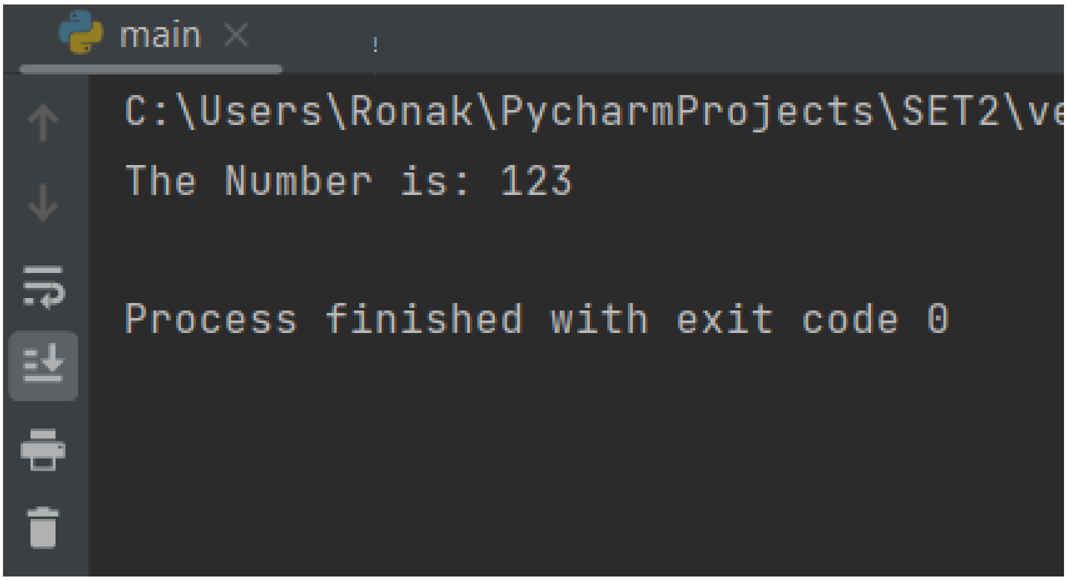
```
C:\Users\Ronak\PycharmProjects\Set1\venv\Scripts\pytho
Enter the value of Base:43
Enter the value of Height:253
The area is 5439.5

Process finished with exit code 0
```

Practical: 6

Aim: Write a python program to display maximum of given 3 numbers.

```
x=5

y=7

z=123

if(x>=y)and(x>=z):

   largest=x

elif(y>=x)and(y>=z):

   largest=y

else:

   largest=z

print("The Number is:",largest)
```

Output:



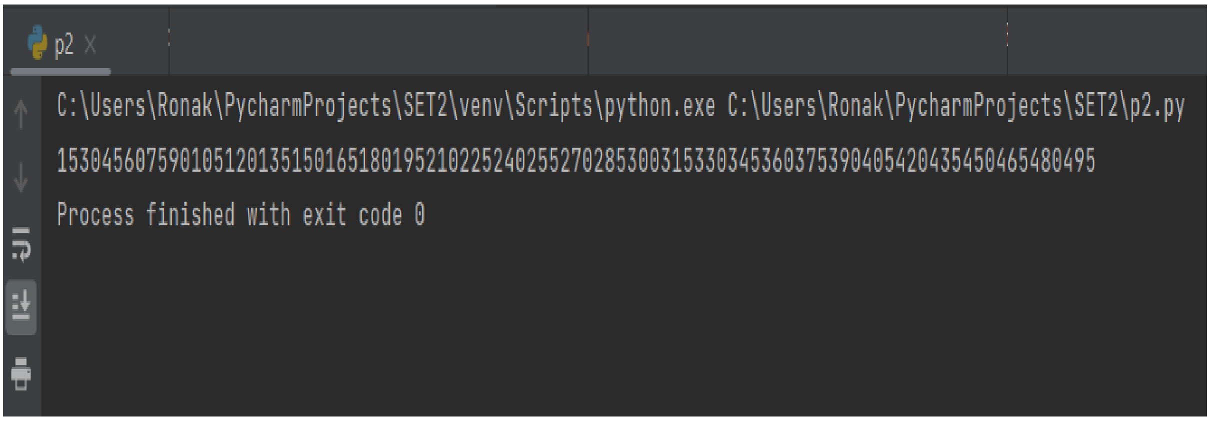
```
C:\Users\Ronak\PycharmProjects\SET2\ve
The Number is: 123

Process finished with exit code 0
```

Practical: 7

Aim: Write a python program to find those numbers which are divisible by 3 and multiple of 5 within 500.

```
for x in range(1,501):
    if(x % 3==0 and x % 5==0):
        print(x, end='')
```

Output:



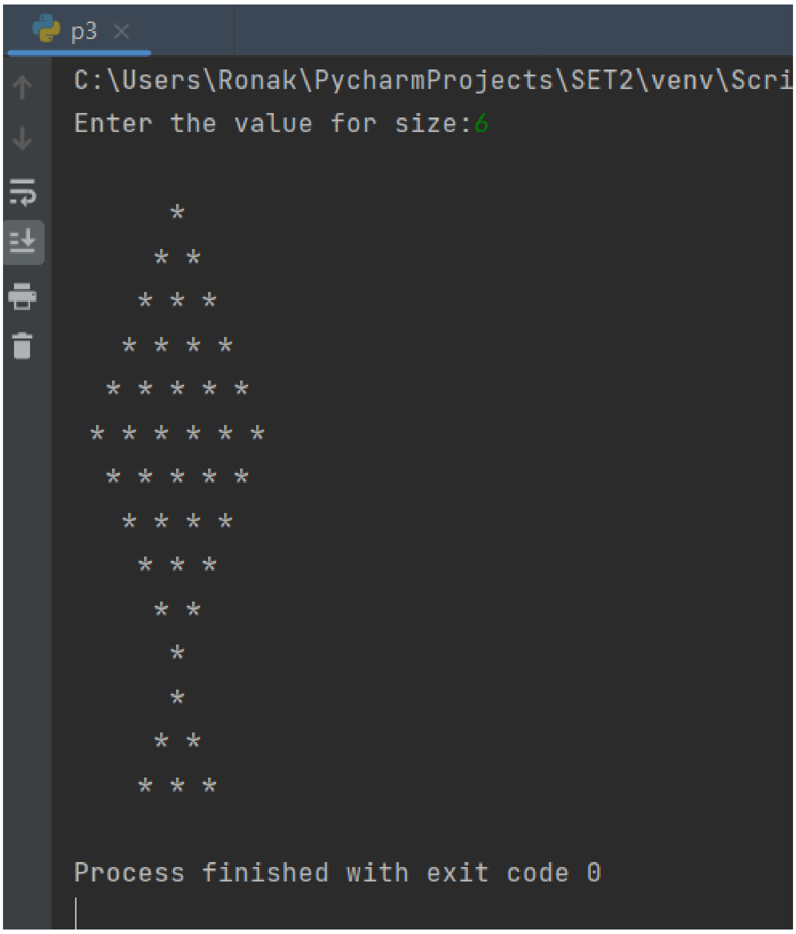
```
C:\Users\Ronak\PycharmProjects\SET2\venv\Scripts\python.exe C:\Users\Ronak\PycharmProjects\SET2\p2.py
153045607590105120135150165180195210225240255270285300315330345360375390405420435450465480495
Process finished with exit code 0
```

Practical: 8

Aim: Write a python program to draw kite pattern using for loop.

```
r=int(input("Enter the value for size:"))
for x in range(r,0,-1):
   print(" "*x,"* "*(r-x))
for x in range(0,r,1):
   print(" "*x,"* "*(r-x))
for x in range(r-1,int(r/2)-1,-1):
   print(" "*x,"* "*(r-x))
```

Output:



```
C:\Users\Ronak\PycharmProjects\SET2\venv\Scri
Enter the value for size:6

      * 
     * * 
    * * * 
   * * * * 
  * * * * * 
 * * * * * * 
  * * * * * 
   * * * * 
    * * * 
     * * 
      * 
      * 
     * * 
    * * * 

Process finished with exit code 0
```

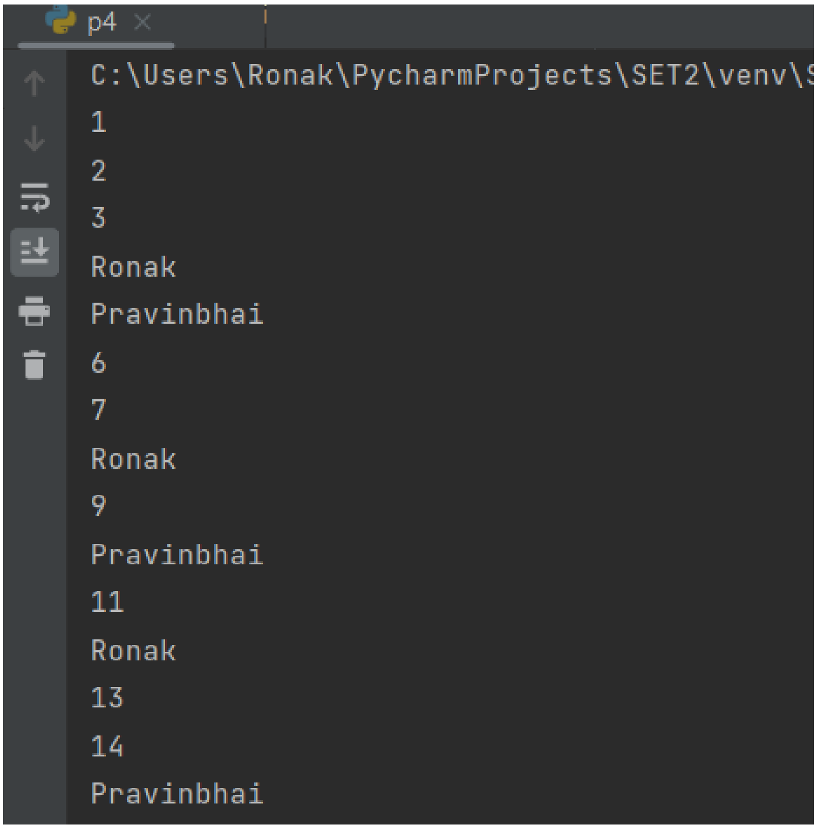
Practical Set-2: Looping and Data Structure with Python

Practical: 1

Aim: Write a python program to print numbers from 1 to 50. For multiple of 4 print name instead of number and for multiple of 5 print father name. For the numbers which are multiple of both 4 and 5 print surname.

```
for i in range(1,51):
    b=i
    if(i%4==0):
        b="Ronak"
    if(i%5==0):
        b="Pravinbhai"
    if(i%4==0 and i%5==0):
        b="Patel"
    print(b)
```

Output:



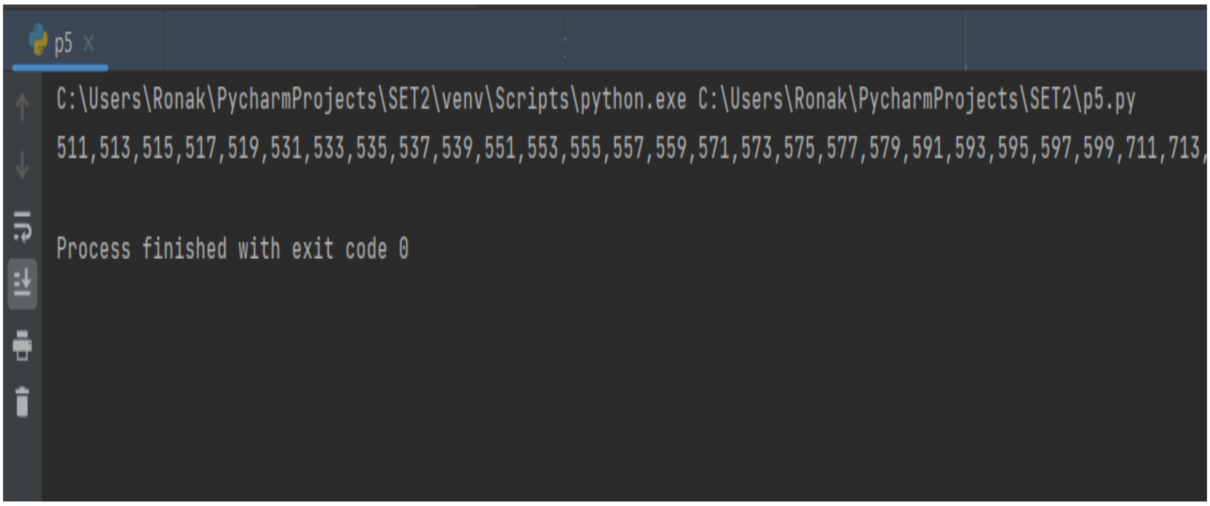
```
C:\Users\Ronak\PycharmProjects\SET2\venv\S
1
2
3
Ronak
Pravinbhai
6
7
Ronak
9
Pravinbhai
11
Ronak
13
14
Pravinbhai
```

Practical: 2

Aim: Write a python program to find numbers between 500 and 800 when each digit of number is ODD and the number should be printed in sequence separated by comma.

```
item=[]
for i in range(500,801):
    s=str(i)
    if(int(s[0])%2!=0) and (int(s[1])%2!=0) and (int(s[2])%2!=0):
        item.append(s)
print(",".join(item))
```

Output:



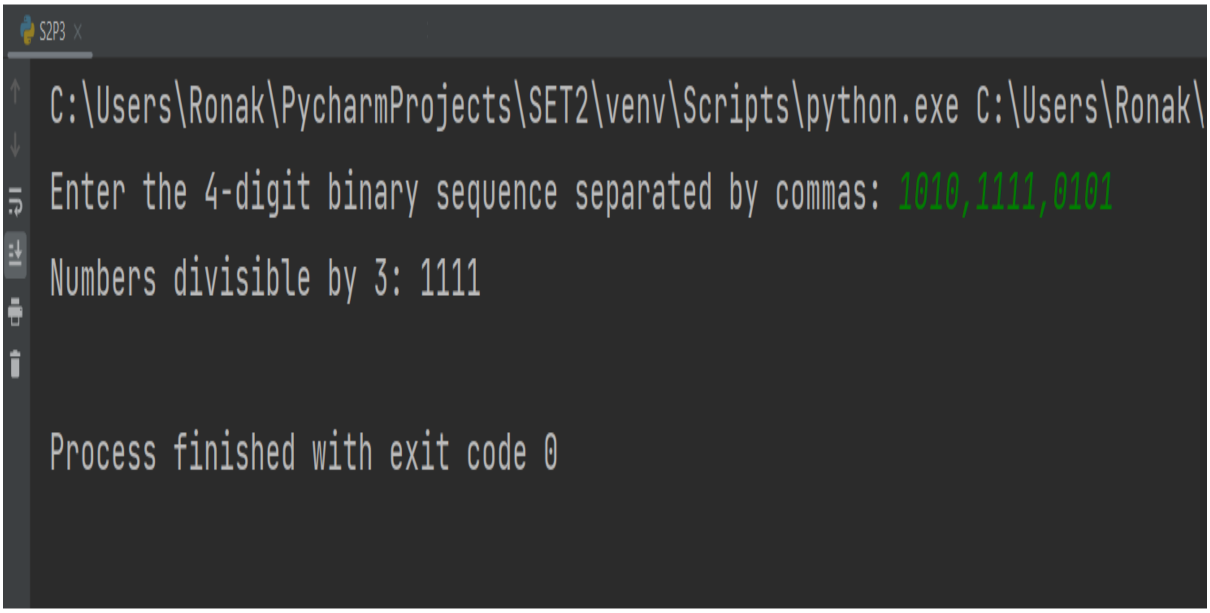
```
C:\Users\Ronak\PycharmProjects\SET2\venv\Scripts\python.exe C:\Users\Ronak\PycharmProjects\SET2\p5.py
511,513,515,517,519,531,533,535,537,539,551,553,555,557,559,571,573,575,577,579,591,593,595,597,599,711,713,

Process finished with exit code 0
```

Practical: 3

Aim: Write a python program which accept a sequence of 4-digit binary numbers separated by comma and also print the numbers which are divisible by 3 in sequence separated by comma.

```
def is_divisible_by_3(binary_str):
    decimal_value = int(binary_str, 2)
    return decimal_value % 3 == 0

def main():
    binary_sequence = input("Enter the 4-digit binary sequence separated by commas: ")
    binary_numbers = binary_sequence.split(",")

    divisible_by_3 = [binary for binary in binary_numbers if is_divisible_by_3(binary)]

    if divisible_by_3:
        print("Numbers divisible by 3:", ", ".join(divisible_by_3))
    else:
        print("No numbers in the sequence are divisible by 3.")

if __name__ == "__main__":
    main()
```

Output:



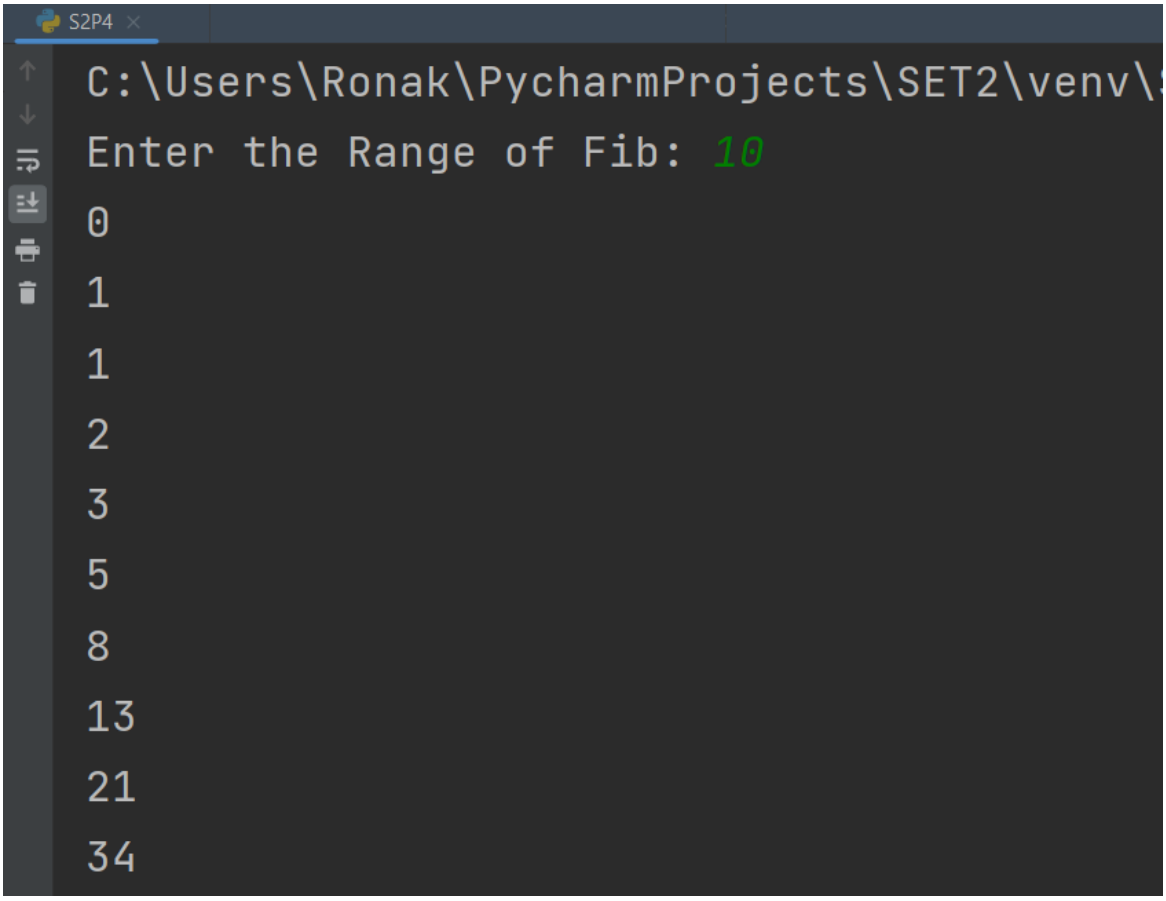
```
C:\Users\Ronak\PycharmProjects\SET2\venv\Scripts\python.exe C:\Users\Ronak\
Enter the 4-digit binary sequence separated by commas: 1010,1111,0101
Numbers divisible by 3: 1111


Process finished with exit code 0
```

Practical: 4

Aim: Write a python program to display Fibonacci sequence up to nth term using recursive functions.

```
def recurr_fib(n):
    if n <= 1:
        return n
    else:
        return recurr_fib(n - 1) + recurr_fib(n - 2)

num = int(input("Enter the Range of Fib: "))

if num <= 0:
    print("Enter a positive Number")
else:
    for i in range(num):
        print(recurr_fib(i))
```

Output:



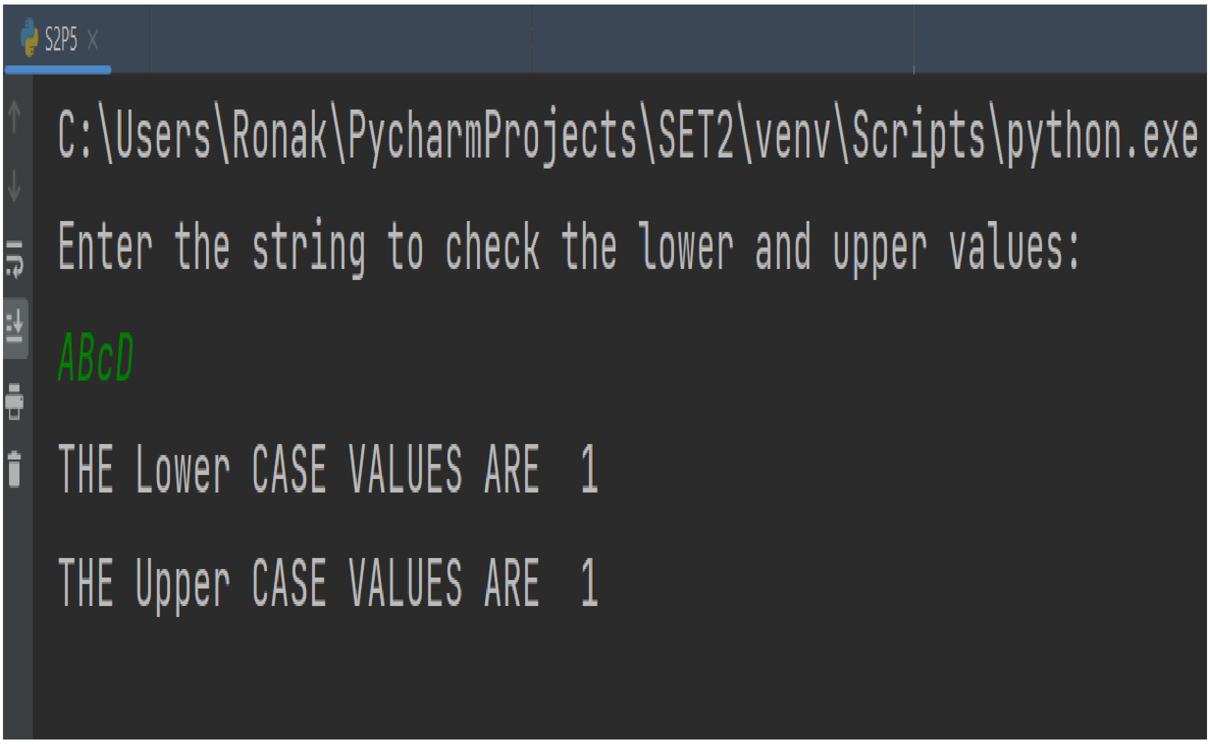
```
C:\Users\Ronak\PycharmProjects\SET2\venv\
Enter the Range of Fib: 10
0
1
1
2
3
5
8
13
21
34
```

Practical: 5

Aim: Write a python program that accept a string and calculate the number of uppercase and lowercase letter.

```
str=input("Enter the string to check the lower and upper values: \n")
b=0
m=0
for i in str:
    if(i<='z' and i>='a'):
        b=b+1
if( i>='A' and i<='Z'):
    m=m+1
print("THE Lower CASE VALUES ARE ",b)
print("THE Upper CASE VALUES ARE ",m)
```

Output:



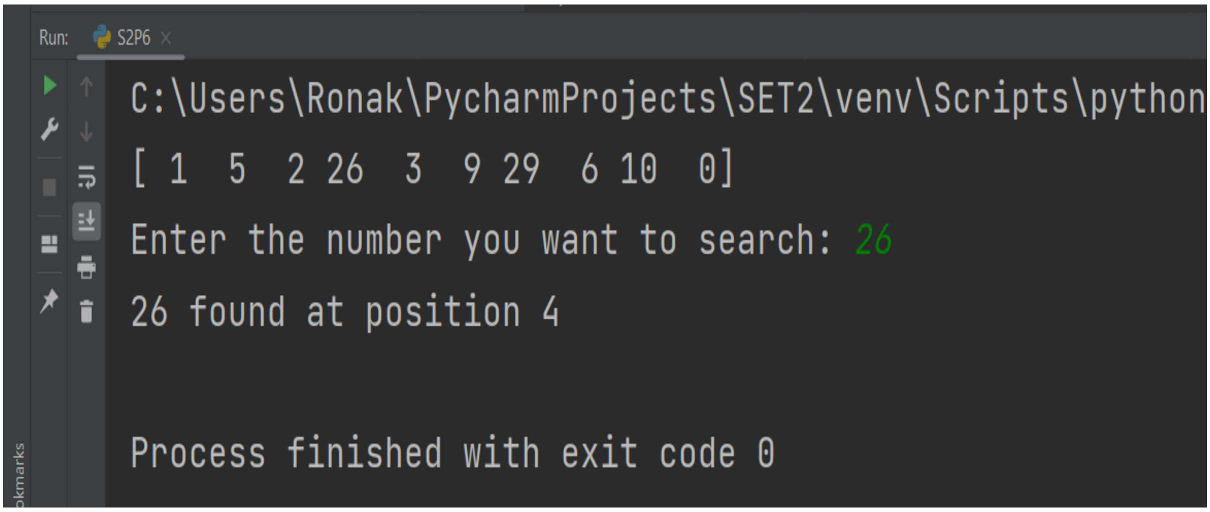
```
C:\Users\Ronak\PycharmProjects\SET2\venv\Scripts\python.exe
Enter the string to check the lower and upper values:
ABcD
THE Lower CASE VALUES ARE  1
THE Upper CASE VALUES ARE  1
```

Practical: 6

Aim: Write a python program to search a number in array using sequential search.

```
import numpy as np

def seq_search(array, num):
    pos = 0
    found = False
    while pos < len(array) and not found:
        if array[pos] == num:
            found = True
        else:
            pos = pos + 1
    return found, pos + 1

array1 = np.random.randint(50, size=(10))
print(array1)
number = int(input("Enter the number you want to search: "))
result, position = seq_search(array1, number)

if result:
    print(f"{number} found at position {position}")
else:
    print(f"{number} not found in the array.")
```

Output:



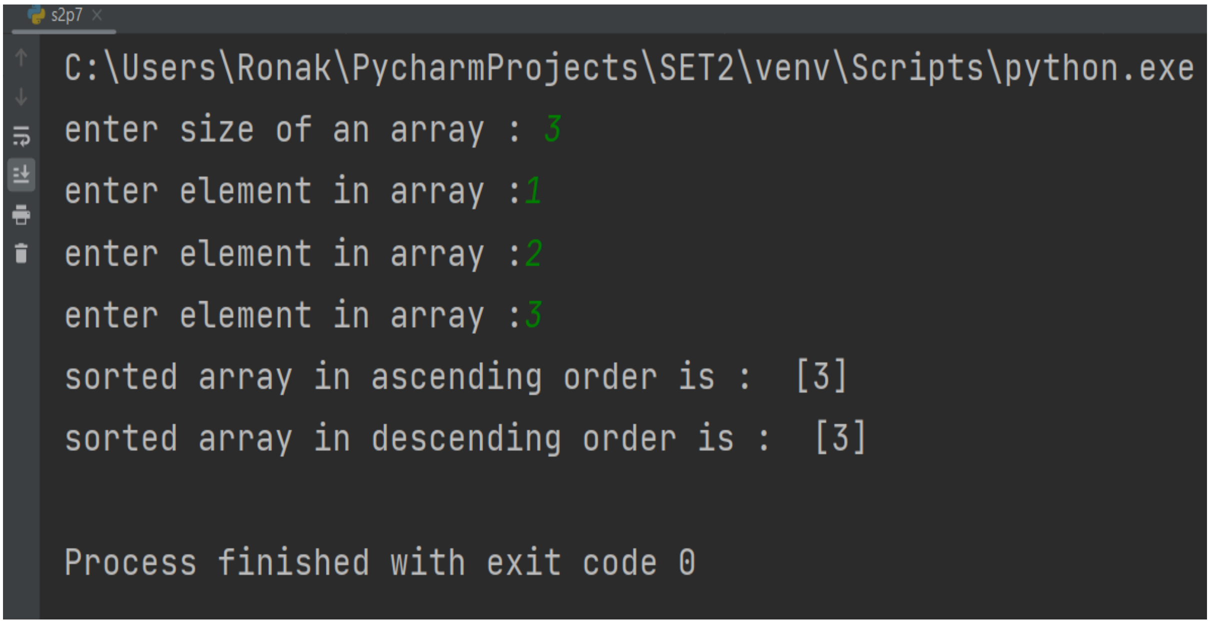
```
C:\Users\Ronak\PycharmProjects\SET2\venv\Scripts\python
[ 1  5  2 26  3  9 29  6 10  0]
Enter the number you want to search: 26
26 found at position 4

Process finished with exit code 0
```

Practical: 7

Aim: Write a python program to sort elements of array.

```
size = int(input("enter size of an array : "))
arr = []
for i in range(size):
    element = int(input("enter element in array :"))
arr.append(element)
arr.sort()
print("sorted array in ascending order is : ",arr)
print("sorted array in descending order is : ",arr[::-1])
```

Output:



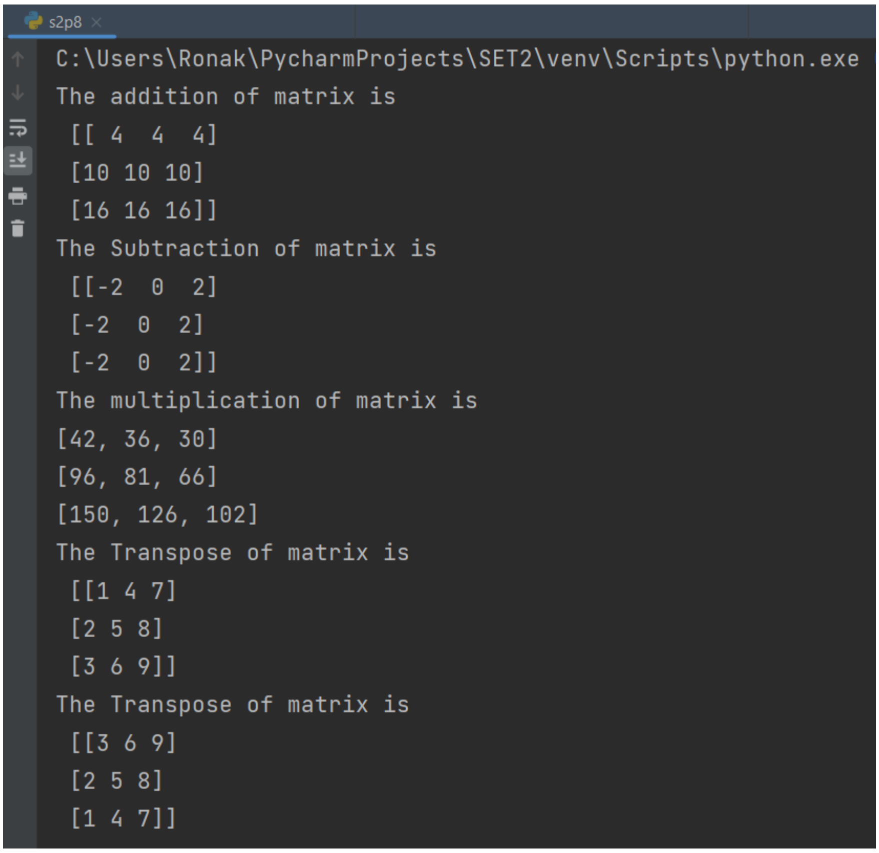
```
C:\Users\Ronak\PycharmProjects\SET2\venv\Scripts\python.exe
enter size of an array : 3
enter element in array :1
enter element in array :2
enter element in array :3
sorted array in ascending order is :  [3]
sorted array in descending order is :  [3]

Process finished with exit code 0
```

Practical: 8

Aim: Write a python program to input two matrix and perform the following given operation:

```
import numpy
arr1=([1,2,3],[4,5,6],[7,8,9])
arr2=([3,2,1],[6,5,4],[9,8,7])
result=([0,0,0],[0,0,0],[0,0,0])
mat1=numpy.array(arr1)
mat2=numpy.array(arr2)
print("The addition of matrix is \n",mat1+mat2)
print("The Subtraction of matrix is \n",mat1-mat2)
for i in range(0,len(mat1),1):
    for k in range(0,len(mat2[0]),1):
        for j in range(0,len(mat2),1):
            result[i][j] +=mat1[i][k]*mat2[k][j]
print("The multiplication of matrix is ",)
for m in result:
    print(m)
print("The Transpose of matrix is \n",mat1.T)
print("The Transpose of matrix is \n",mat2.T)
```

Output:



```
C:\Users\Ronak\PycharmProjects\SET2\venv\Scripts\python.exe
The addition of matrix is
 [[ 4  4  4]
 [10 10 10]
 [16 16 16]]
The Subtraction of matrix is
 [[-2  0  2]
 [-2  0  2]
 [-2  0  2]]
The multiplication of matrix is
[42, 36, 30]
[96, 81, 66]
[150, 126, 102]
The Transpose of matrix is
 [[1 4 7]
 [2 5 8]
 [3 6 9]]
The Transpose of matrix is
 [[3 6 9]
 [2 5 8]
 [1 4 7]]
```

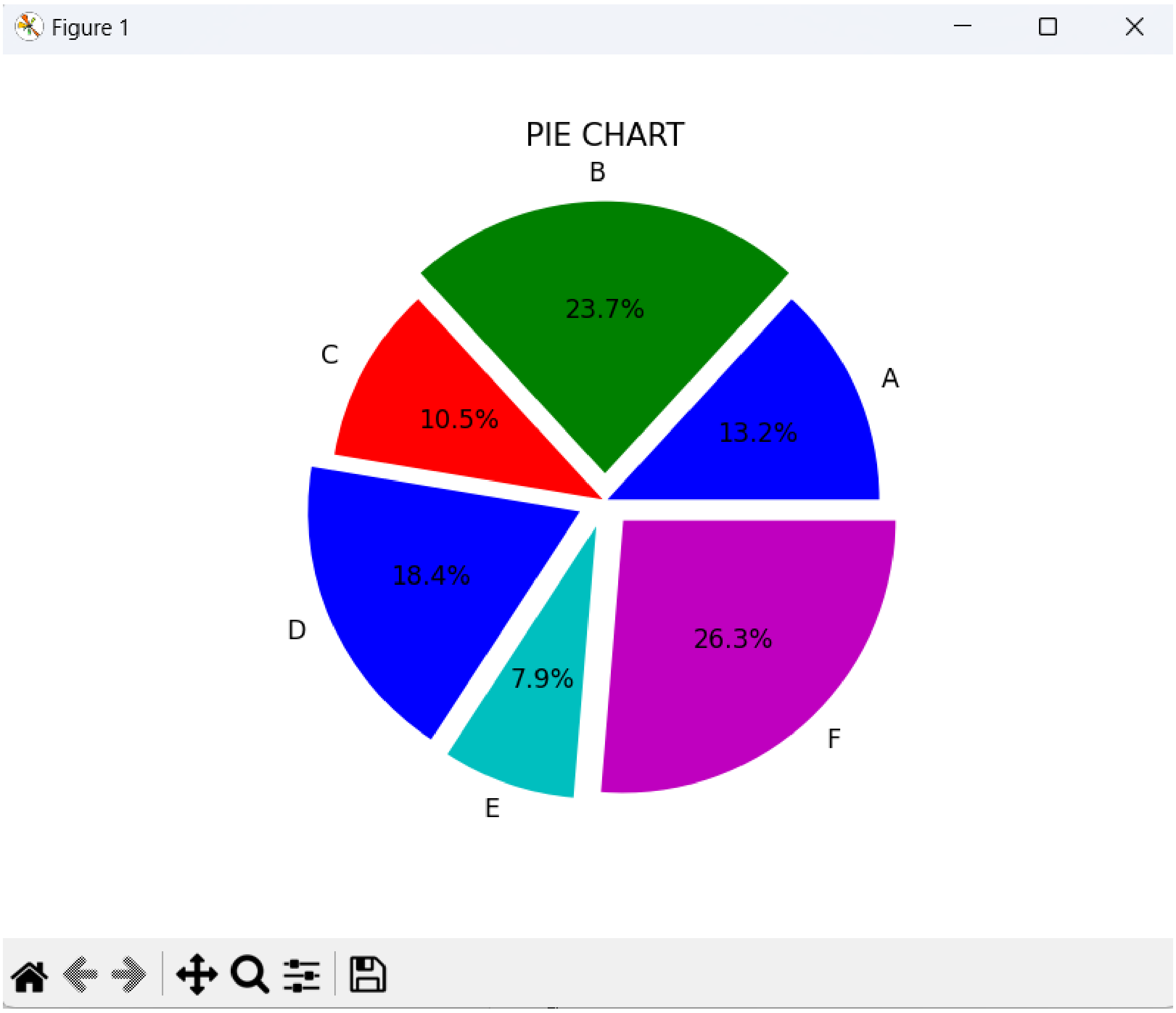
Practical Set-6: Create various plots using matplotlib library.

Practical: 1

Aim: Prepare a Pie charts by taking suitable data as reference.

```
import matplotlib.pyplot as plt
#%matplotlib inline
values = [5,9,4,7,3,10]
c = ['b','g','r','b','c','m']
l = ['A','B','C','D','E','F']
e = (0.01,0.1,0.01,0.1,0.1,0.1)
plt.pie(values,colors=c,labels=l,explode=e,autopct='%1.1f%%',shadow=False)
plt.title('PIE CHART');
plt.show()
```

Output:

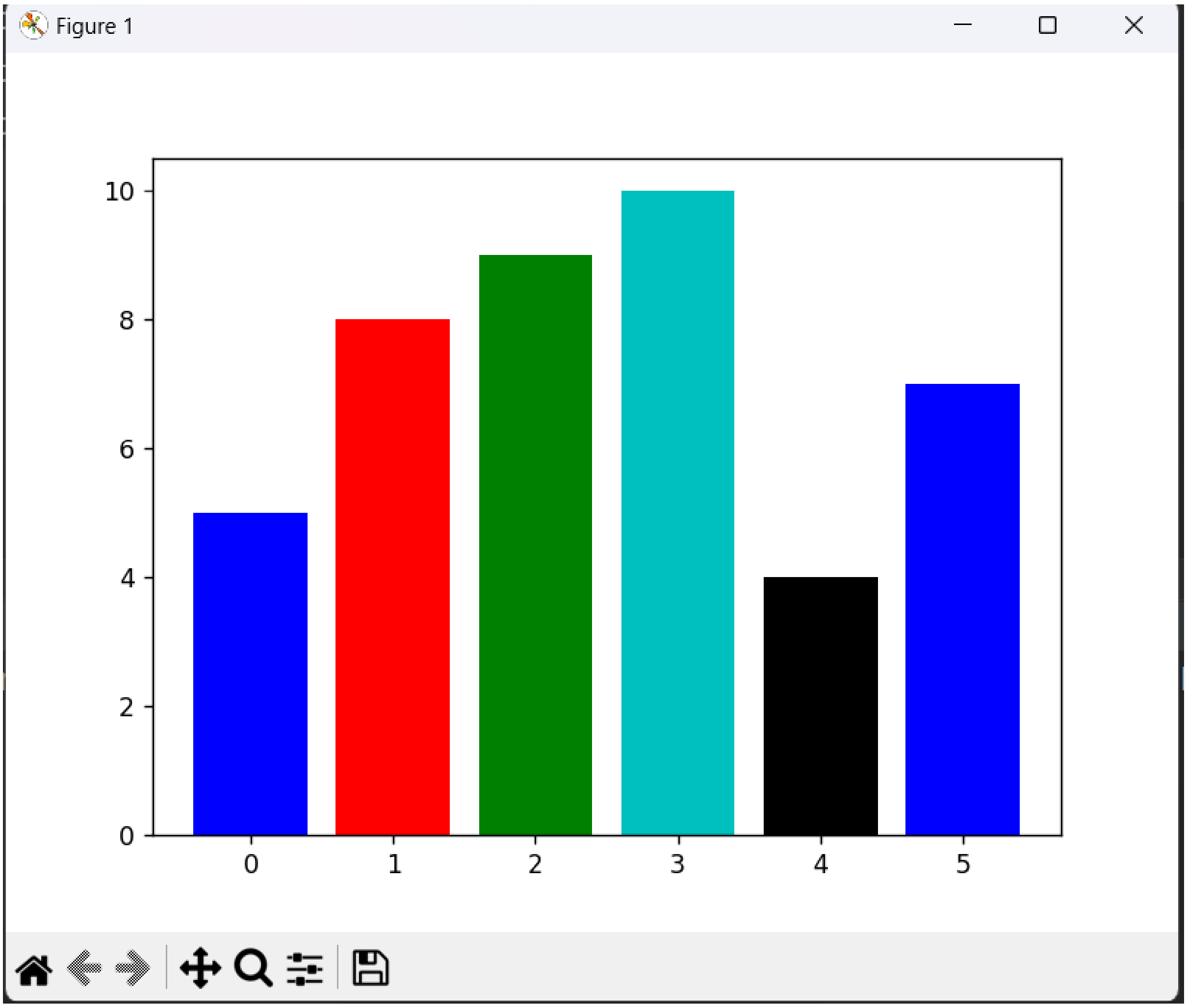


Practical: 2

Aim: Prepare a Bar charts by taking suitable data as reference.

```
import matplotlib.pyplot as plt
#%matplotlib inline
x=[5,8,9,10,4,7]
y=[0.7,0.8,0.7,0.7,0.8,0.7]
colors = ['b','r','g','c','k','b']
plt.bar(range(0,6),x,color=colors,align='center')
plt.show()
```

Output:

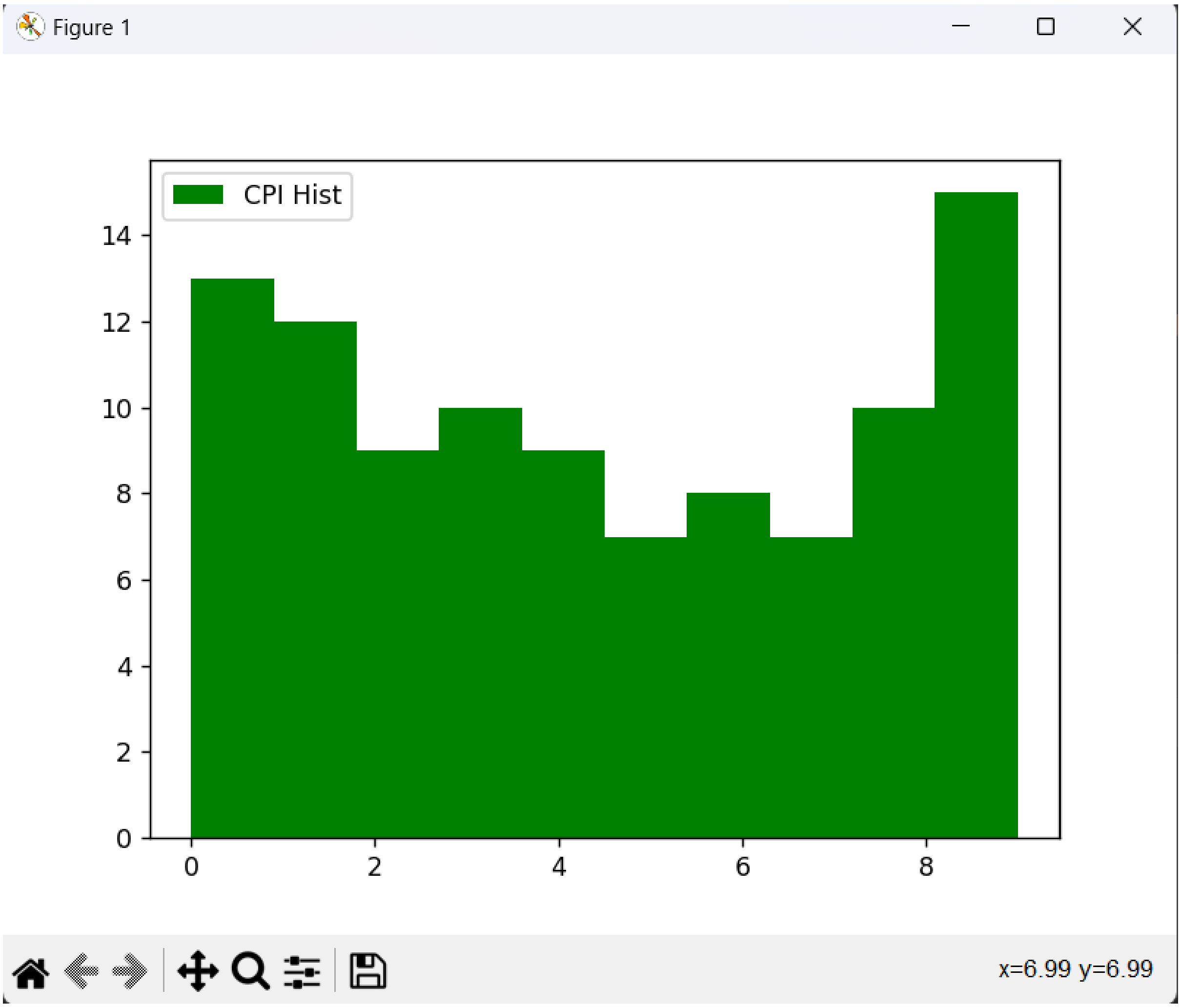


Practical: 3

Aim: Prepare a Histograms by taking suitable data as reference.

```
import matplotlib.pyplot as plt

import numpy as np

#%matplotlib notebook

cpis = np.random.randint(0,10,100)

plt.hist(cpis,bins=10,histtype='stepfilled',align='mid',label='CPI Hist',color='g')

plt.legend()

plt.show()
```

Output:

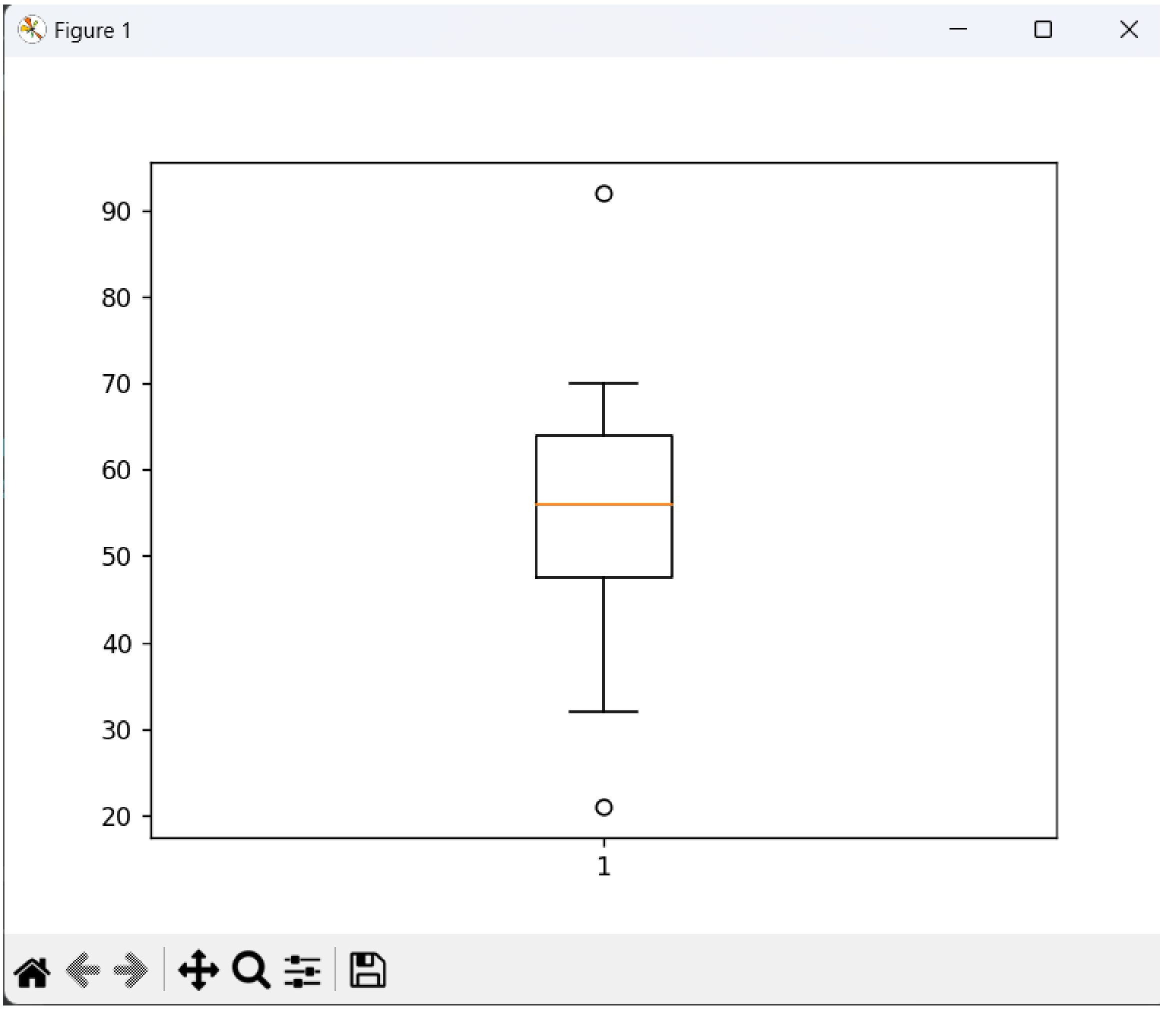


Practical: 4

Aim: Prepare a Box plots by taking suitable data as reference.

```
import pandas as pd

import matplotlib.pyplot as plt

timetaken =pd.Series([50,45,52,63,70,21,56,68,54,57,35,62,65,92,32])

plt.boxplot(timetaken)
plt.show()
```

Output:



Practical: 5

Aim: Prepare a Scatterplots by taking suitable data as reference.

```
import matplotlib.pyplot as plt
import pandas as pd
df=pd.read_csv('insurance.csv')
grouped = df.groupby(['Smoker'])
for key, group in grouped:
    plt.scatter(group['bmi'],
    group['Charges'],
    label='Smoke ='+ str(key))

plt.legend()
plt.show()
```

Insurance.csv

bmi	Smoker	Charges
27.9	Yes	16884.9
33.77	No	1725.55
33	No	4449.46
22.705	No	21984.5
28.88	No	3866.86
25.74	No	3756.62
33.44	No	8240.59
27.74	Yes	7281.51
29.83	Yes	6406.41
25.84	No	28923.1
22.4	Yes	36120.9
30.2	Yes	40273.7
28.5	No	1137.01
34.1	Yes	43921.2
31.9	Yes	38282.8
28.6	No	3238.44
24.7	Yes	39611.8
22.3	No	2322.62
36.9	Yes	38511.6
32.2	Yes	43993.8
28.1	No	2234.38
30.6	Yes	40932.4
23.9	No	2128.43
36.7	Yes	38571.9
31.5	Yes	39722.8
27.3	No	3258.96
26.2	No	2897.32
25.1	No	2464.62
23.2	No	2349.43
30.1	Yes	36837.5

Output:

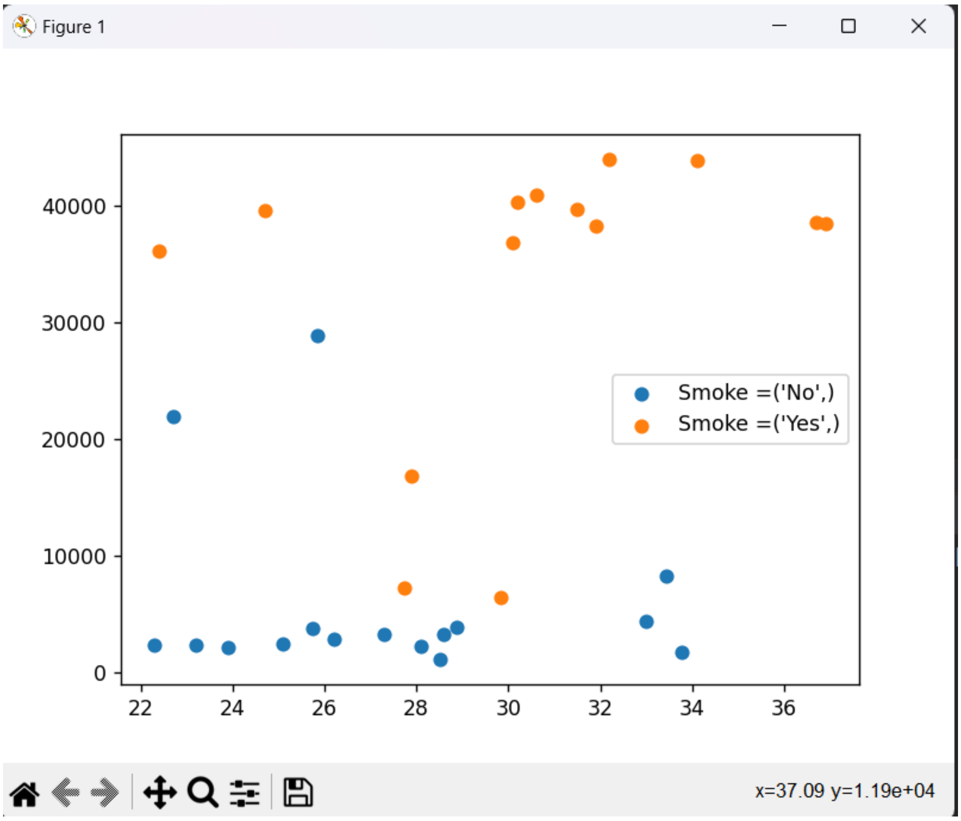


Figure 1
40000
30000
20000
10000
0
22
24
26
28
30
32
34
36
● Smoke =('No',)
● Smoke =('Yes',)
x=37.09 y=1.19e+04

Practical: 6

Aim: Prepare a Time Series by taking suitable data as reference.

import pandas as pd

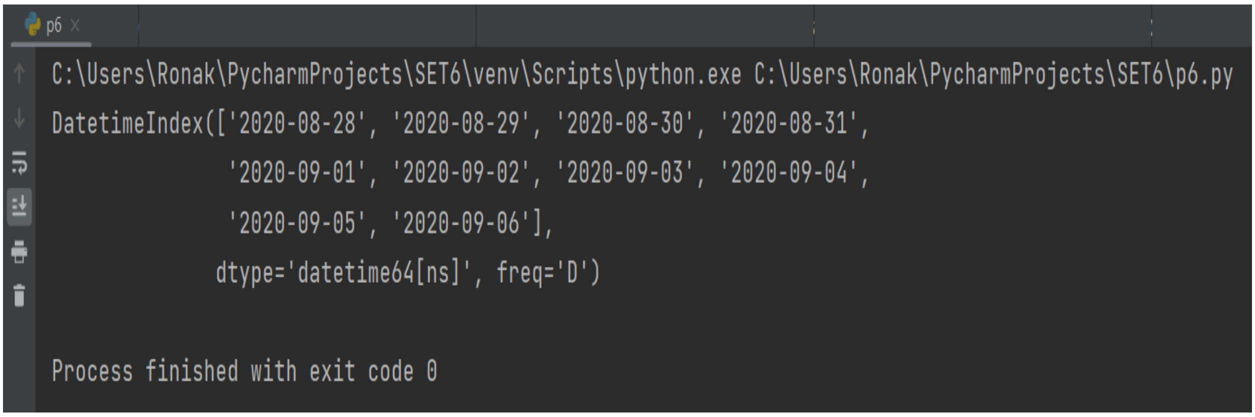
import datetime as dt

start_date = dt.datetime(2020,8,28)

daterange = pd.date_range(start_date,freq='D',periods=10)

print(daterange)

Output:



```
C:\Users\Ronak\PycharmProjects\SET6\venv\Scripts\python.exe C:\Users\Ronak\PycharmProjects\SET6\p6.py
DatetimeIndex(['2020-08-28', '2020-08-29', '2020-08-30', '2020-08-31',
               '2020-09-01', '2020-09-02', '2020-09-03', '2020-09-04',
               '2020-09-05', '2020-09-06'],
              dtype='datetime64[ns]', freq='D')

Process finished with exit code 0
```