DATA SCIENCE & MACHINE LEARNING:

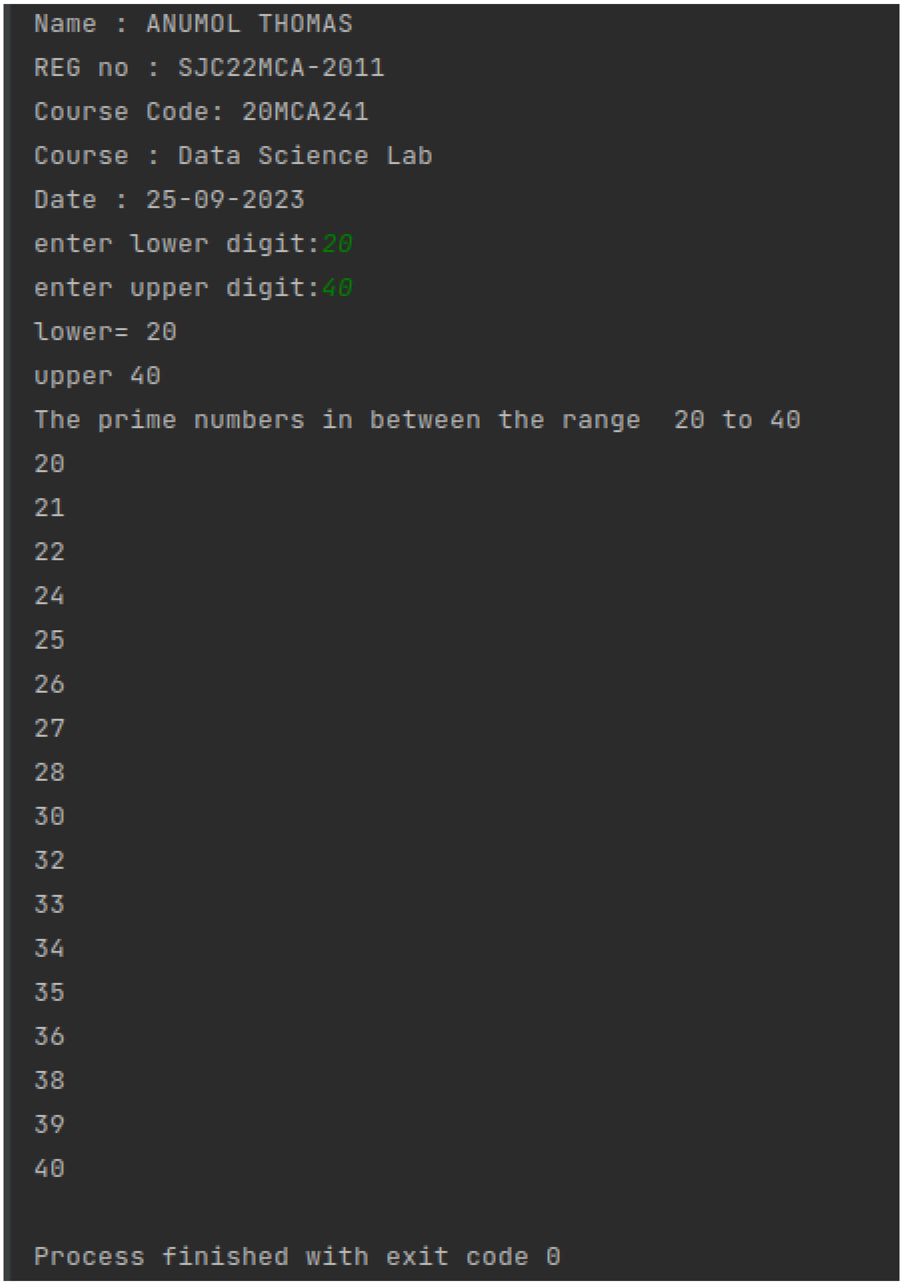
LAB CYCLE 1

1. Program to Print all non-Prime Numbers in an Interval

CODE:-

```
print("Name : ANUMOL THOMAS")
print("REG no : SJC22MCA-2011")
print("Course Code: 20MCA241")
print("Course : Data Science Lab")
print("Date : 25-09-2023")
x=int(input("enter lower digit:"))
y=int(input("enter upper digit:"))
print("lower=",x)
print("upper",y)
print("The prime numbers in between the range ",x,"to",y)
for n in range(x, y+1):
    if(n > 1):
        for i in range(2,n):
            if(n % i) == 0:
                print(n)
                break
```

OUTPUT:-



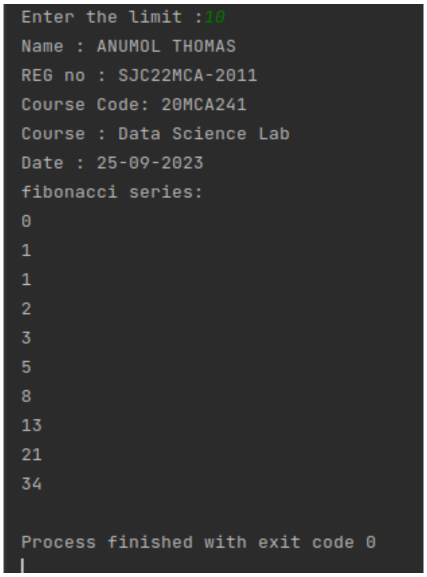
```
Name : ANUMOL THOMAS
REG no : SJC22MCA-2011
Course Code: 20MCA241
Course : Data Science Lab
Date : 25-09-2023
enter lower digit:20
enter upper digit:40
lower= 20
upper 40
The prime numbers in between the range  20 to 40
20
21
22
24
25
26
27
28
30
32
33
34
35
36
38
39
40

Process finished with exit code 0
```

2. Program to print the first N Fibonacci numbers.

CODE:-

```
n=int(input("Enter the limit :"))
n1,n2=0,1
count=0
print("Name : ANUMOL THOMAS")
print("REG no : SJC22MCA-2011")
print("Course Code: 20MCA241")
print("Course : Data Science Lab")
print("Date : 25-09-2023")
if n <= 0:
   print("Invalid")
else:
   print("fibonacci series:")
   while count < n:
      print(n1)
      next=n1+n2
      n1=n2
      n2=next
      count +=1
```

OUTPUT:-



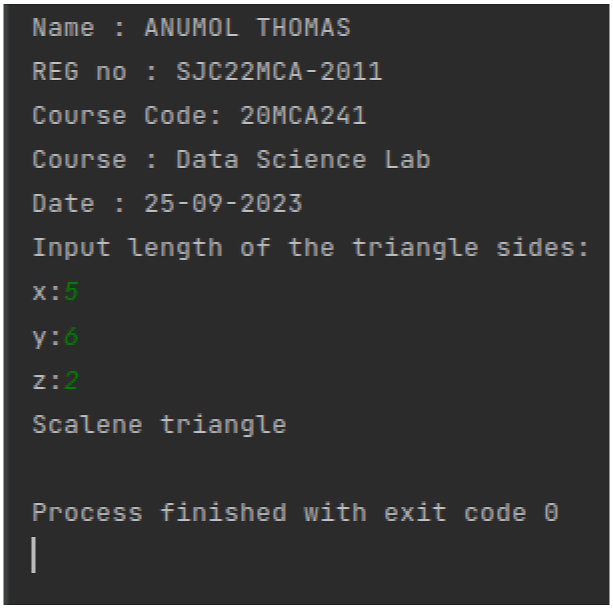
```
Enter the limit :10
Name : ANUMOL THOMAS
REG no : SJC22MCA-2011
Course Code: 20MCA241
Course : Data Science Lab
Date : 25-09-2023
fibonacci series:
0
1
1
2
3
5
8
13
21
34

Process finished with exit code 0
```

3. Given sides of a triangle, write a program to check whether given triangle is an isosceles, equilateral or scalene.

CODE:-

```
print("Name : ANUMOL THOMAS")
print("REG no : SJC22MCA-2011")
print("Course Code: 20MCA241")
print("Course : Data Science Lab")
print("Date : 25-09-2023")
print("Input length of the triangle sides:")
x=int(input("x:"))
y=int(input("y:"))
z=int(input("z:"))
if x==y==z:
   print("Equilateral triangle")
elif x==y or y==z or z==x:
   print("Isosceles triangle")
else:
   print("Scalene triangle")
```

OUTPUT:-



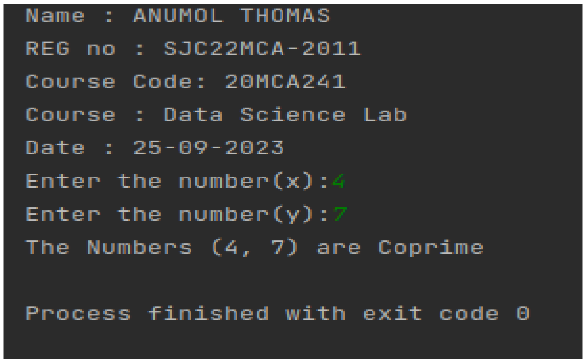
```
Name : ANUMOL THOMAS
REG no : SJC22MCA-2011
Course Code: 20MCA241
Course : Data Science Lab
Date : 25-09-2023
Input length of the triangle sides:
x:5
y:6
z:2
Scalene triangle

Process finished with exit code 0
```

4. Program to check whether given pair of number is coprime

CODE:-

```
def gcd(x,y):
    if(x==0 or y==0):
        return 0
    if(x==y):
        return x
    if(x>y):
        return gcd(x-y,y)
    return gcd(x,y-x)
def coprime(x,y):
    if(gcd(x,y)==1):
        print("The Numbers",(x,y),"are Coprime")
    else:
        print("The Numbers", (x, y), "are not Coprime")
print("Name : ANUMOL THOMAS")
print("REG no : SJC22MCA-2011")
print("Course Code: 20MCA241")
print("Course : Data Science Lab")
print("Date : 25-09-2023")
x=int(input("Enter the number(x):"))
y=int(input("Enter the number(y):"))
coprime(x,y)
```

OUTPUT:-



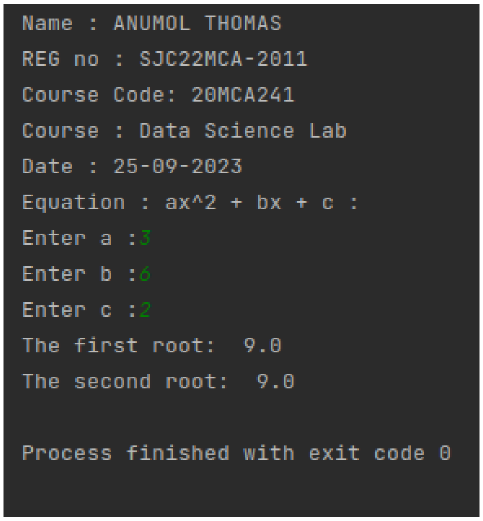
```
Name : ANUMOL THOMAS
REG no : SJC22MCA-2011
Course Code: 20MCA241
Course : Data Science Lab
Date : 25-09-2023
Enter the number(x):4
Enter the number(y):7
The Numbers (4, 7) are Coprime

Process finished with exit code 0
```

5. Program to find the roots of a quadratic equation(rounded to 2 decimal places)

CODE:-

```
print("Name : ANUMOL THOMAS")
print("REG no : SJC22MCA-2011")
print("Course Code: 20MCA241")
print("Course : Data Science Lab")
print("Date : 25-09-2023")
print("Equation : ax^2 + bx + c :")
a=int(input("Enter a :"))
b=int(input("Enter b :"))
c=int(input("Enter c :"))
d=b*b-4*a*c
if(d<0):
    print("The roots are imaginary")
else:
    r1=(-b+d)/2*a
    r2=(-b+d)/2*a
    print("The first root: ",round(r1,2))
    print("The second root: ",round(r2,2))
```

OUTPUT:-



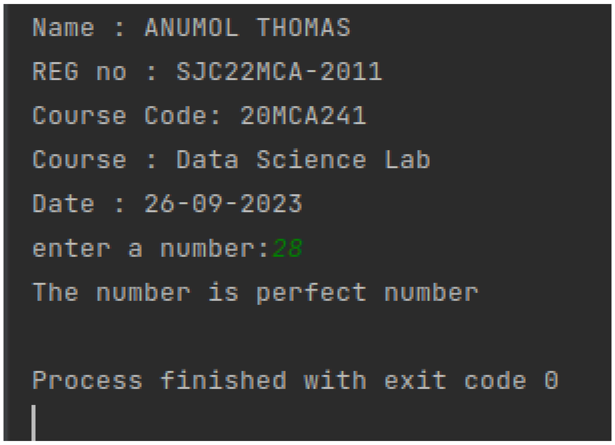
```
Name : ANUMOL THOMAS
REG no : SJC22MCA-2011
Course Code: 20MCA241
Course : Data Science Lab
Date : 25-09-2023
Equation : ax^2 + bx + c :
Enter a :3
Enter b :6
Enter c :2
The first root:  9.0
The second root:  9.0

Process finished with exit code 0
```

6. Program to check whether a given number is perfect number or not(sum of factors=number)

CODE:-

```
print("Name : ANUMOL THOMAS")
print("REG no : SJC22MCA-2011")
print("Course Code: 20MCA241")
print("Course : Data Science Lab")
print("Date : 26-09-2023")
n=int(input("enter a number:"))
sum=0
for i in range(1,n):
    if(n%i==0):
        sum=sum+i
if(sum==n):
    print("The number is perfect number")
else:
    print("The number is not perfect number")
```

OUTPUT:-



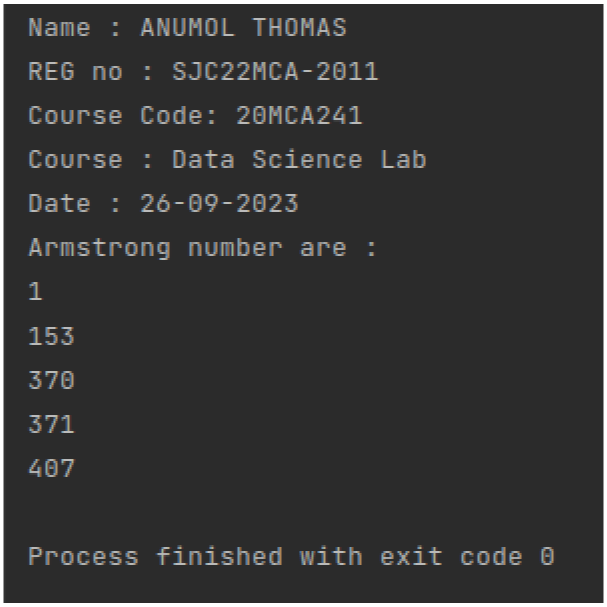
```
Name : ANUMOL THOMAS
REG no : SJC22MCA-2011
Course Code: 20MCA241
Course : Data Science Lab
Date : 26-09-2023
enter a number:28
The number is perfect number

Process finished with exit code 0
```

7. Program to display amstrong numbers upto 1000

CODE:-

```
print("Name : ANUMOL THOMAS")
print("REG no : SJC22MCA-2011")
print("Course Code: 20MCA241")
print("Course : Data Science Lab")
print("Date : 26-09-2023")
print("Armstrong number are :")
for i in range(1,1000):
    a=i
    sum=0
    while(i>0):
        r=i%10
        sum=sum+(r*r*r)
        i=i//10
    if(sum==a):
        print(a)
```

OUTPUT:-



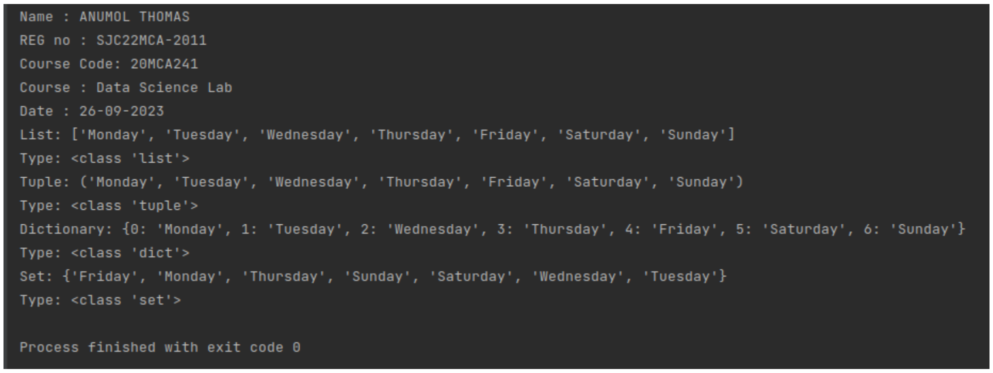
```
Name : ANUMOL THOMAS
REG no : SJC22MCA-2011
Course Code: 20MCA241
Course : Data Science Lab
Date : 26-09-2023
Armstrong number are :
1
153
370
371
407

Process finished with exit code 0
```

8. Store and display the days of a week as a List, Tuple, Dictionary, Set. Also demonstrate different ways to store values in each of them. Display its type also.

CODE:-

```
print("Name : ANUMOL THOMAS")
print("REG no : SJC22MCA-2011")
print("Course Code: 20MCA241")
print("Course : Data Science Lab")
print("Date : 26-09-2023")
days_list = ["Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"]
print("List:", days_list)
print("Type:", type(days_list))
days_tuple = ("Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday")
print("Tuple:", days_tuple)
print("Type:", type(days_tuple))
days_dict = {0: "Monday", 1: "Tuesday", 2: "Wednesday", 3: "Thursday", 4: "Friday", 5: "Saturday", 6: "Sunday"}
print("Dictionary:", days_dict)
print("Type:", type(days_dict))
days_set = {"Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"}
print("Set:", days_set)
print("Type:", type(days_set))
```

OUTPUT:-



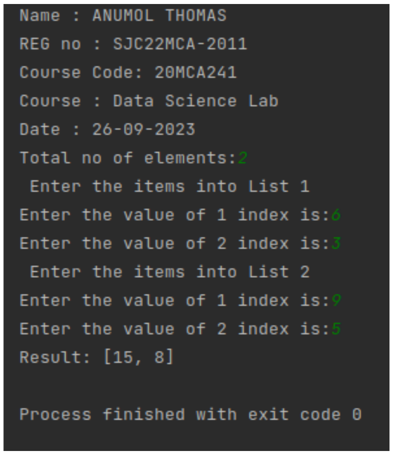
```
Name : ANUMOL THOMAS
REG no : SJC22MCA-2011
Course Code: 20MCA241
Course : Data Science Lab
Date : 26-09-2023
List: ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday']
Type: <class 'list'>
Tuple: ('Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday')
Type: <class 'tuple'>
Dictionary: {0: 'Monday', 1: 'Tuesday', 2: 'Wednesday', 3: 'Thursday', 4: 'Friday', 5: 'Saturday', 6: 'Sunday'}
Type: <class 'dict'>
Set: {'Friday', 'Monday', 'Thursday', 'Sunday', 'Saturday', 'Wednesday', 'Tuesday'}
Type: <class 'set'>

Process finished with exit code 0
```

9. Write a program to add elements of given 2 lists

CODE:-

```
l1=[]
l2=[]
l3=[]
print("Name : ANUMOL THOMAS")
print("REG no : SJC22MCA-2011")
print("Course Code: 20MCA241")
print("Course : Data Science Lab")
print("Date : 26-09-2023")
n=int(input("Total no of elements:"))
print (" Enter the items into List 1  ")
for i in range(1,n+1):
   num=int(input("Enter the value of %d index is:" %i))
   l1.append(num)
print (" Enter the items into List 2  ")
for i in range(1,n+1):
   num=int(input("Enter the value of %d index is:" %i))
   l2.append(num)
for j in range(n):
   l3.append(l1[j]+l2[j])
print("Result:",l3)
```

OUTPUT:-



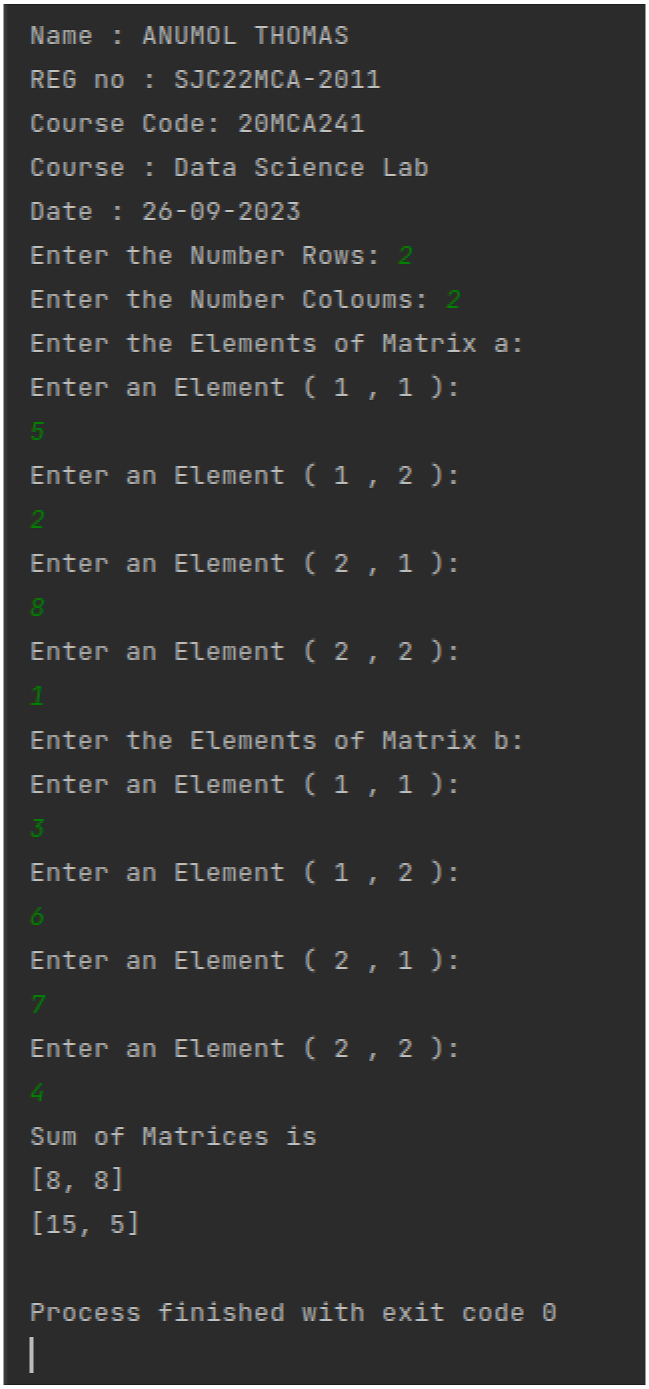
```
Name : ANUMOL THOMAS
REG no : SJC22MCA-2011
Course Code: 20MCA241
Course : Data Science Lab
Date : 26-09-2023
Total no of elements:2
 Enter the items into List 1
Enter the value of 1 index is:6
Enter the value of 2 index is:3
 Enter the items into List 2
Enter the value of 1 index is:9
Enter the value of 2 index is:5
Result: [15, 8]

Process finished with exit code 0
```

10. Write a program to find the sum of 2 matrices using nested List.

CODE:-

```
print("Name : ANUMOL THOMAS")
print("REG no : SJC22MCA-2011")
print("Course Code: 20MCA241")
print("Course : Data Science Lab")
print("Date : 26-09-2023")
r=int(input("Enter the Number Rows: "));
c=int(input("Enter the Number Coloums: "));
a = [[0 for j in range(0,c)] for i in range(0,r)]
b = [[0 for j in range(0,c)] for i in range(0,r)]
sum = [[0 for j in range(0,c)] for i in range(0,r)]
print("Enter the Elements of Matrix a:")
for i in range(0,r):
    for j in range(0,c):
        print("Enter an Element (",i+1,",",j+1,"): ")
        a[i][j]= int(input())
print("Enter the Elements of Matrix b:")
for i in range(0,r):
    for j in range(0,c):
        print("Enter an Element (",i+1,",",j+1,"): ")
        b[i][j]=int(input())
for i in range(0,r):
    for j in range(0,c):
        sum[i][j]=a[i][j]+b[i][j]

print("Sum of Matrices is")
for i in sum:
    print(i)
```

OUTPUT:-



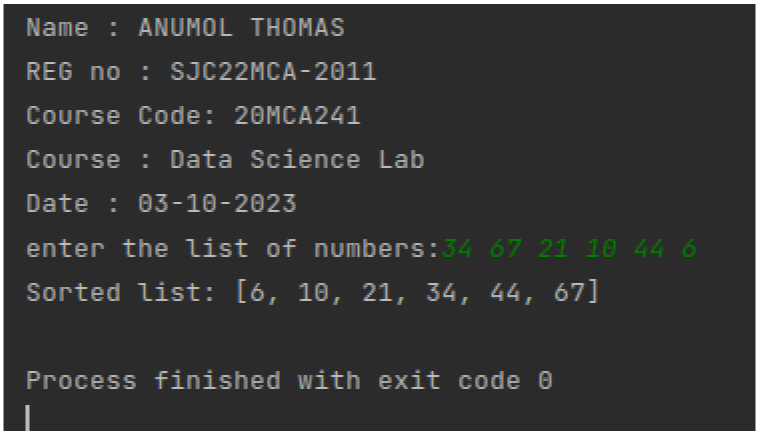
```
Name : ANUMOL THOMAS
REG no : SJC22MCA-2011
Course Code: 20MCA241
Course : Data Science Lab
Date : 26-09-2023
Enter the Number Rows: 2
Enter the Number Coloums: 2
Enter the Elements of Matrix a:
Enter an Element ( 1 , 1 ):
5
Enter an Element ( 1 , 2 ):
2
Enter an Element ( 2 , 1 ):
8
Enter an Element ( 2 , 2 ):
1
Enter the Elements of Matrix b:
Enter an Element ( 1 , 1 ):
3
Enter an Element ( 1 , 2 ):
6
Enter an Element ( 2 , 1 ):
7
Enter an Element ( 2 , 2 ):
4
Sum of Matrices is
[8, 8]
[15, 5]

Process finished with exit code 0
```

11. Write a program to perform bubble sort on a given set of elements.

CODE:-

```
def bubble_sort(l):
    for i in range(len(l) - 1,0,-1):
        no_swap=True
        for j in range(0,i):
            if l[j+1] < l[j]:
                l[j],l[j+1]=l[j+1],l[j]
                no_swap=False
        if no_swap:
            return
print("Name : ANUMOL THOMAS")
print("REG no : SJC22MCA-2011")
print("Course Code: 20MCA241")
print("Course : Data Science Lab")
print("Date : 03-10-2023")
l=input("enter the list of numbers:").split()
l=[int (x) for x in l]
bubble_sort(l)
print("Sorted list: ",end="")
print(l)
```

OUTPUT:-



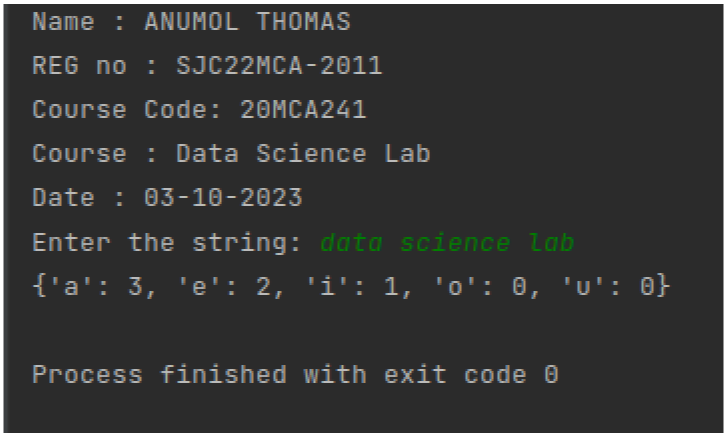
```
Name : ANUMOL THOMAS
REG no : SJC22MCA-2011
Course Code: 20MCA241
Course : Data Science Lab
Date : 03-10-2023
enter the list of numbers:34 67 21 10 44 6
Sorted list: [6, 10, 21, 34, 44, 67]

Process finished with exit code 0
```

12. Program to find the count of each vowel in a string(use dictionary)

CODE:-

```
print("Name : ANUMOL THOMAS")
print("REG no : SJC22MCA-2011")
print("Course Code: 20MCA241")
print("Course : Data Science Lab")
print("Date : 03-10-2023")
vowels='aeiou'
a=input("Enter the string:")
a=a.casefold()
count={}.fromkeys(vowels,0)
for char in a:
   if char in count:
      count[char] +=1
print(count)
```

OUTPUT:-



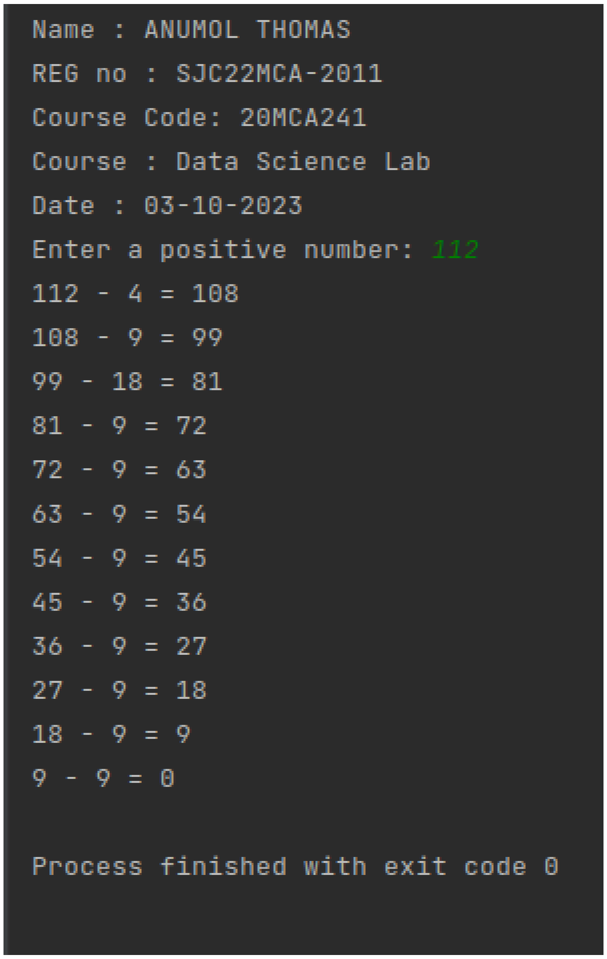
```
Name : ANUMOL THOMAS
REG no : SJC22MCA-2011
Course Code: 20MCA241
Course : Data Science Lab
Date : 03-10-2023
Enter the string: data science lab
{'a': 3, 'e': 2, 'i': 1, 'o': 0, 'u': 0}

Process finished with exit code 0
```

13. Write a Python program that accept a positive number and subtract from this number the sum of its digits and so on. Continues this operation until the number is positive(eg: 256-&gt;2+5+6=13
256-13=243
243-9=232……..

CODE:-

```
print("Name : ANUMOL THOMAS")
print("REG no : SJC22MCA-2011")
print("Course Code: 20MCA241")
print("Course : Data Science Lab")
print("Date : 03-10-2023")
def sum_of_digits(n):
    digit_sum = 0
    while n > 0:
        digit_sum += n % 10
        n //= 10
    return digit_sum
try:
    num = int(input("Enter a positive number: "))
    if num <= 0:
        print("Please enter a positive number.")
    else:
        while num > 0:
            digit_sum = sum_of_digits(num)
            print(f"{num} - {digit_sum} = {num - digit_sum}")
            num -= digit_sum
except ValueError:
    print("Invalid input. Please enter a valid positive number.")
```

OUTPUT:-



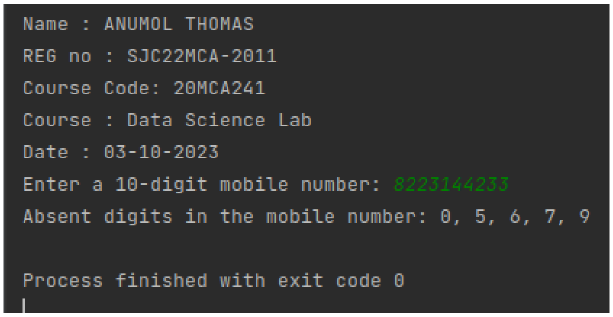
```
Name : ANUMOL THOMAS
REG no : SJC22MCA-2011
Course Code: 20MCA241
Course : Data Science Lab
Date : 03-10-2023
Enter a positive number: 112
112 - 4 = 108
108 - 9 = 99
99 - 18 = 81
81 - 9 = 72
72 - 9 = 63
63 - 9 = 54
54 - 9 = 45
45 - 9 = 36
36 - 9 = 27
27 - 9 = 18
18 - 9 = 9
9 - 9 = 0

Process finished with exit code 0
```

14. Write a Python program that accepts a 10 digit mobile number, and find the digits which are absent in a given mobile number

CODE:-

```
print("Name : ANUMOL THOMAS")
print("REG no : SJC22MCA-2011")
print("Course Code: 20MCA241")
print("Course : Data Science Lab")
print("Date : 03-10-2023")
def find_absent_digits(mobile_number):
    all_digits = set("0123456789")
    mobile_digits = set(mobile_number)
    absent_digits = all_digits - mobile_digits
    return sorted(list(absent_digits))
try:
    mobile_number = input("Enter a 10-digit mobile number: ")
    if len(mobile_number) == 10 and mobile_number.isdigit():
        absent_digits = find_absent_digits(mobile_number)
        if absent_digits:
            print("Absent digits in the mobile number:", ', '.join(absent_digits))
        else:
            print("The mobile number contains all digits from 0 to 9.")
    else:
        print("Invalid input. Please enter a valid 10-digit mobile number.")
except ValueError:
    print("Invalid input. Please enter a valid 10-digit mobile number.")
```

OUTPUT:-



```
Name : ANUMOL THOMAS
REG no : SJC22MCA-2011
Course Code: 20MCA241
Course : Data Science Lab
Date : 03-10-2023
Enter a 10-digit mobile number: 8223144233
Absent digits in the mobile number: 0, 5, 6, 7, 9

Process finished with exit code 0
```