Name   : Siddhi Parekh

Reg No. : 221071047

Batch   : C

SY CE

Experiment 1:

A] Square root of a number

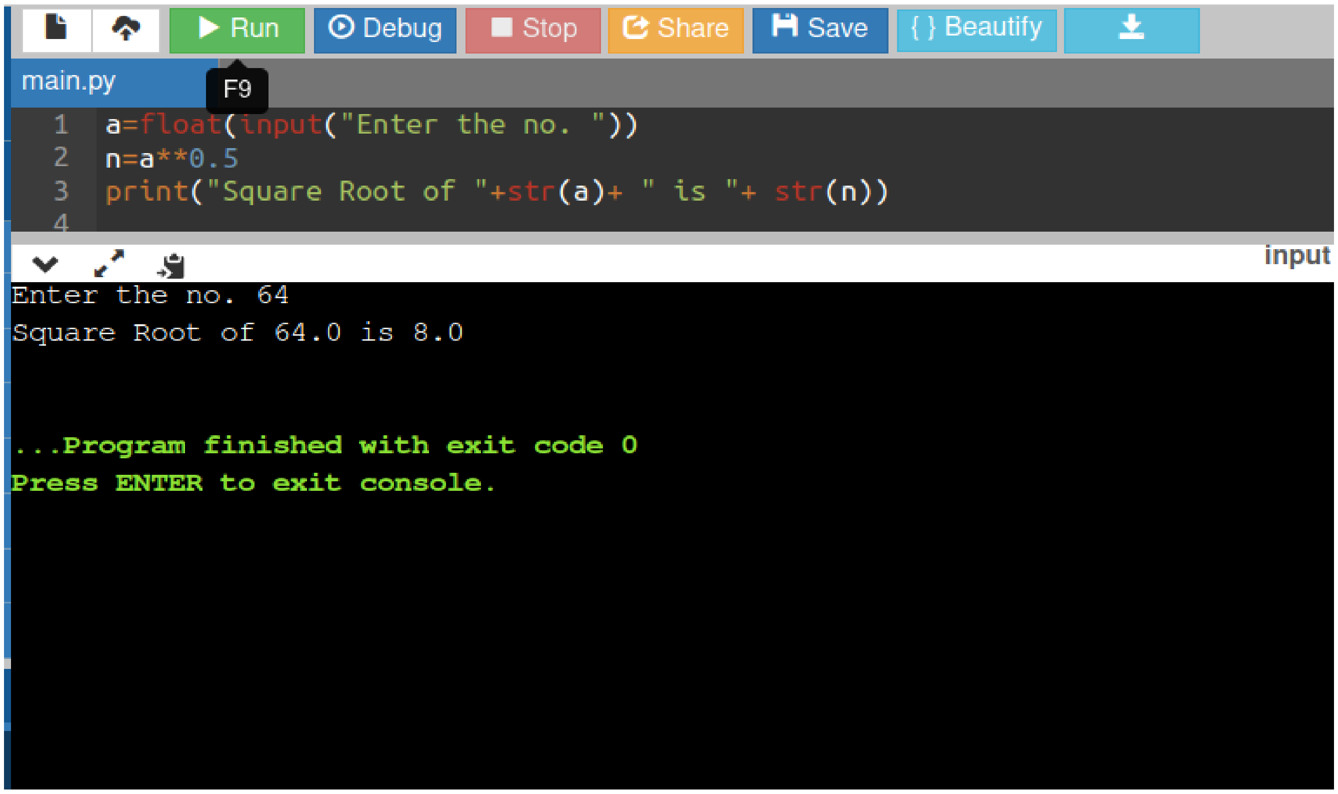
AIM :To find square root of a number

THEORY: Square root of a number is a value, which on multiplication by itself, gives the original number.

Here we have used the ** operator.
Double asterisks (**) acts as an exponentiation operator for numeric values.

CODE:

```
a=float(input("Enter the no. "))
n=a**0.5
print("Square Root of "+str(a)+ " is "+ str(n))
```



```python
a=float(input("Enter the no. "))
n=a**0.5
print("Square Root of "+str(a)+ " is "+ str(n))
```

```
Enter the no. 64
Square Root of 64.0 is 8.0


...Program finished with exit code 0
Press ENTER to exit console.
```

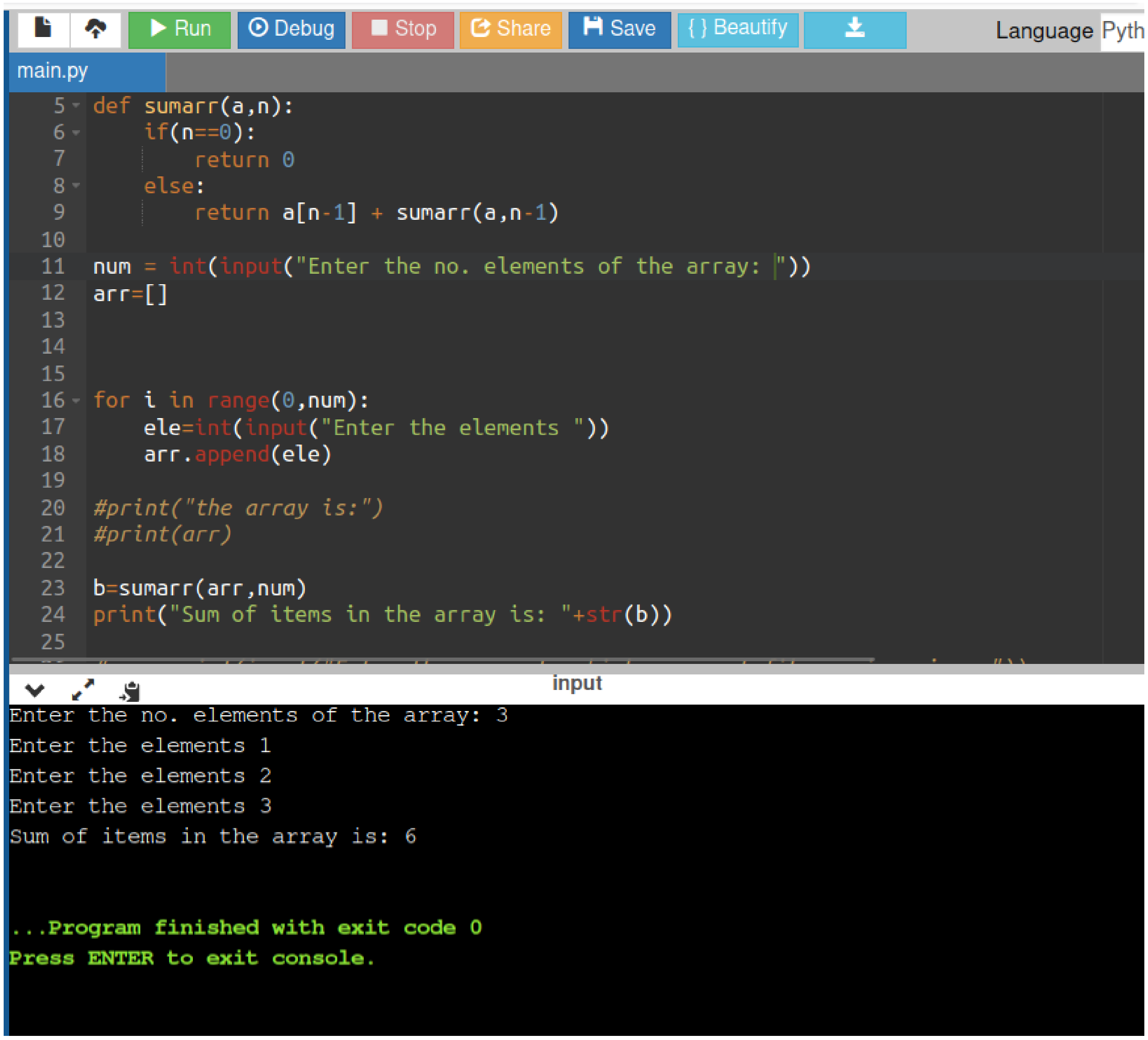
B] Sum of elements of array using recursion

AIM: To find sum of elements of an Array using Recursion

Theory: Recursion is the process of defining something in terms of itself.
In simple words, it is a process in which a function calls itself directly or indirectly.
Every recursive function must have a base condition that stops the recursion or else the function calls itf infinitely.

Code:

```
def sumarr(a,n):
    if(n==0):
        return 0
    else:
        return a[n-1] + sumarr(a,n-1)

num = int(input("Enter the no. elements of the array: "))
arr=[]



for i in range(0,num):
        ele=int(input("Enter the elements "))
        arr.append(ele)

#print("the array is:")
#print(arr)

b=sumarr(arr,num)
print("Sum of items in the array is: "+str(b))
```



```python
def sumarr(a,n):
    if(n==0):
        return 0
    else:
        return a[n-1] + sumarr(a,n-1)

num = int(input("Enter the no. elements of the array: "))
arr=[]



for i in range(0,num):
    ele=int(input("Enter the elements "))
    arr.append(ele)

#print("the array is:")
#print(arr)

b=sumarr(arr,num)
print("Sum of items in the array is: "+str(b))

```

```
Enter the no. elements of the array: 3
Enter the elements 1
Enter the elements 2
Enter the elements 3
Sum of items in the array is: 6


...Program finished with exit code 0
Press ENTER to exit console.
```

C] Fibonacci of a number

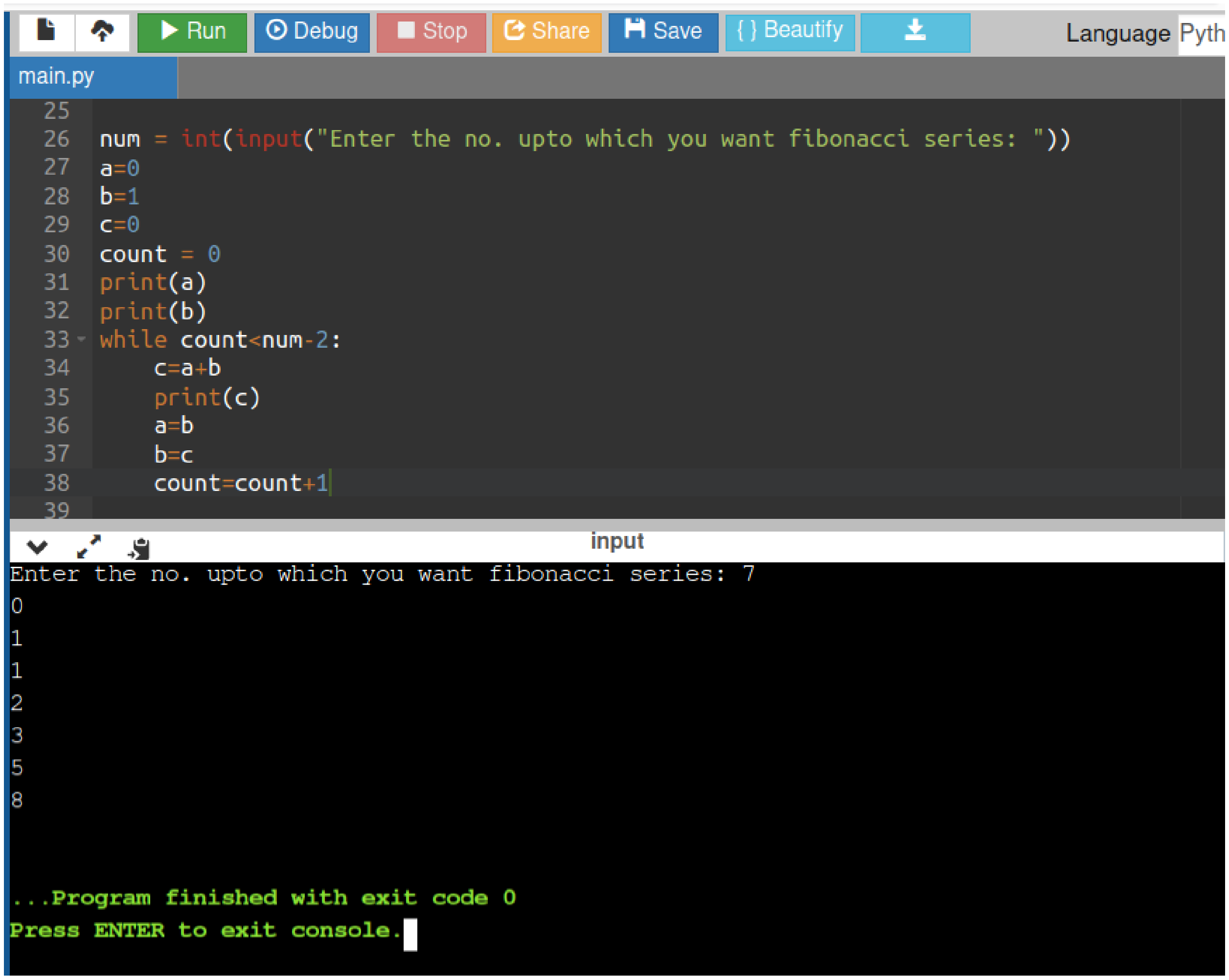
AIM :To find fibonacci of a number

THEORY: The Fibonacci series is the sequence of numbers (also called Fibonacci numbers), where every number is the sum of the preceding two numbers, such that the first two terms are '0' and '1'.

Here, we use a for loop where we update the value and perform addition.

CODE:

```
num = int(input("Enter the no. upto which you want fibonacci series: "))
a=0
b=1
c=0
count = 0
print(a)
print(b)
while count<num-2:
    c=a+b
    print(c)
    a=b
    b=c
    count=count+1
```



```python

num = int(input("Enter the no. upto which you want fibonacci series: "))
a=0
b=1
c=0
count = 0
print(a)
print(b)
while count<num-2:
    c=a+b
    print(c)
    a=b
    b=c
    count=count+1

```

```
Enter the no. upto which you want fibonacci series: 7
0
1
1
2
3
5
8


...Program finished with exit code 0
Press ENTER to exit console.
```

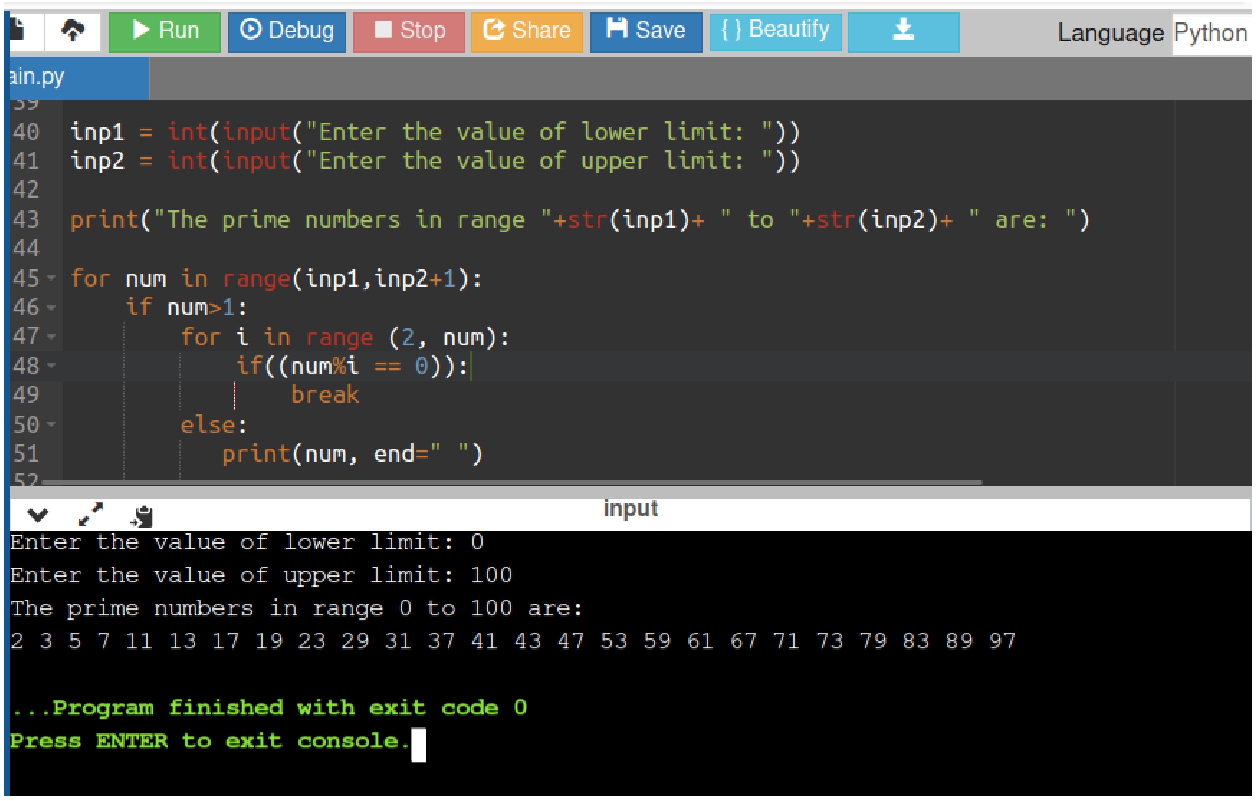
D] Implement service to find prime numbers

AIM: To implement a service to find prime numbers.

THEORY: A prime number is a whole number greater than 1 whose only factors are 1 and itself.
Here to find prime numbers in a given range, we use the for loop and the if-else condition.

CODE:

```
inp1 = int(input("Enter the value of lower limit: "))
inp2 = int(input("Enter the value of upper limit: "))

print("The prime numbers in range "+str(inp1)+ " to "+str(inp2)+ " are: ")

for num in range(inp1,inp2+1):
    if num>1:
        for i in range (2, num):
            if((num%i == 0)):
                break
else:
  print(num, end=" ")
```



```
inp1 = int(input("Enter the value of lower limit: "))
inp2 = int(input("Enter the value of upper limit: "))

print("The prime numbers in range "+str(inp1)+ " to "+str(inp2)+ " are: ")

for num in range(inp1,inp2+1):
    if num>1:
        for i in range (2, num):
            if((num%i == 0)):
                break
        else:
          print(num, end=" ")
```

```
Enter the value of lower limit: 0
Enter the value of upper limit: 100
The prime numbers in range 0 to 100 are:
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

...Program finished with exit code 0
Press ENTER to exit console.
```

CONCLUSION:

Thus, from this experiment 1, python's syntax, its implementation in various ways is learnt. Also,

1.We learnt about power operator, since square root had to be found without using sqrt().

2.We learnt about function recursion for calculating the sum of elements in an array.

3.We learnt about loops for finding the Fibonacci series for a number of terms.

4.We also learnt about looping and conditional statements to find the prime numbers in a given range