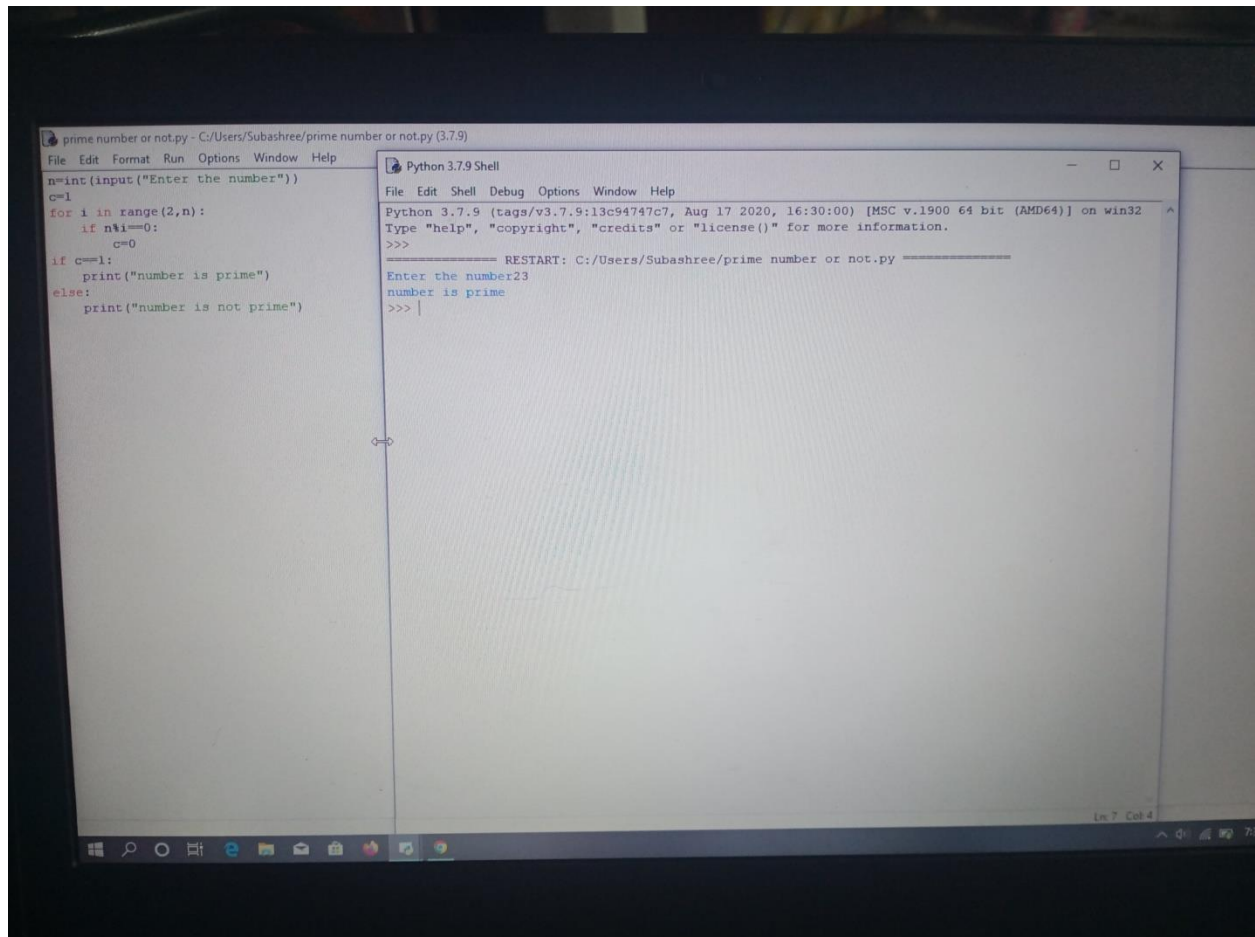


## MODULE PYTHON ASSIGNMENT-1

1. Write a python program to test a given number is prime or not.



The screenshot shows a Python IDE with two windows. The left window, titled 'prime number or not.py - C:/Users/Subashree/prime number or not.py (3.7.9)', contains the following code:

```
n=int(input("Enter the number"))
c=1
for i in range(2,n):
    if n%i==0:
        c=0
if c==1:
    print("number is prime")
else:
    print("number is not prime")
```

The right window, titled 'Python 3.7.9 Shell', shows the execution of the program. It displays the prompt 'Enter the number', the input '23', and the output 'number is prime'.

```
Python 3.7.9 Shell
File Edit Shell Debug Options Window Help
Python 3.7.9 (tags/v3.7.9:13c9474c7, Aug 17 2020, 16:30:00) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Subashree/prime number or not.py =====
Enter the number23
number is prime
>>>
```

prime number or not.py - C:/Users/Subashree/prime number or not.py (3.7.9)

File Edit Format Run Options Window Help

```
n=int(input("Enter the number"))
c=1
for i in range(2,n):
    if n%i==0:
        c=0
if c==1:
    print("number is prime")
else:
    print("number is not prime")
```

Python 3.7.9 Shell

File Edit Shell Debug Options Window Help

Python 3.7.9 (tags/v3.7.9:13c94747c7, Aug 17 2020, 16:30:00) [MSC v.1900 64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license()" for more information.

>>>

===== RESTART: C:/Users/Subashree/prime number or not.py =====

Enter the number4

number is not prime

>>>

Ln 7 Col 4

prime number or not.py - C:/Users/Subashree/prime number or not.py (3.7.9)

File Edit Format Run Options Window Help

```
n=int(input("Enter the number"))
c=1
for i in range(2,n):
    if n%i==0:
        c=0
if c==1:
    print("number is prime")
else:
    print("number is not prime")
```

Python 3.7.9 Shell

File Edit Shell Debug Options Window Help

Python 3.7.9 (tags/v3.7.9:13c94747c7, Aug 17 2020, 16:30:00) [MSC v.1900 64 bit (AMD64)] on win32

Type "help", "copyright", "credits" or "license()" for more information.

>>>

----- RESTART: C:/Users/Subashree/prime number or not.py -----

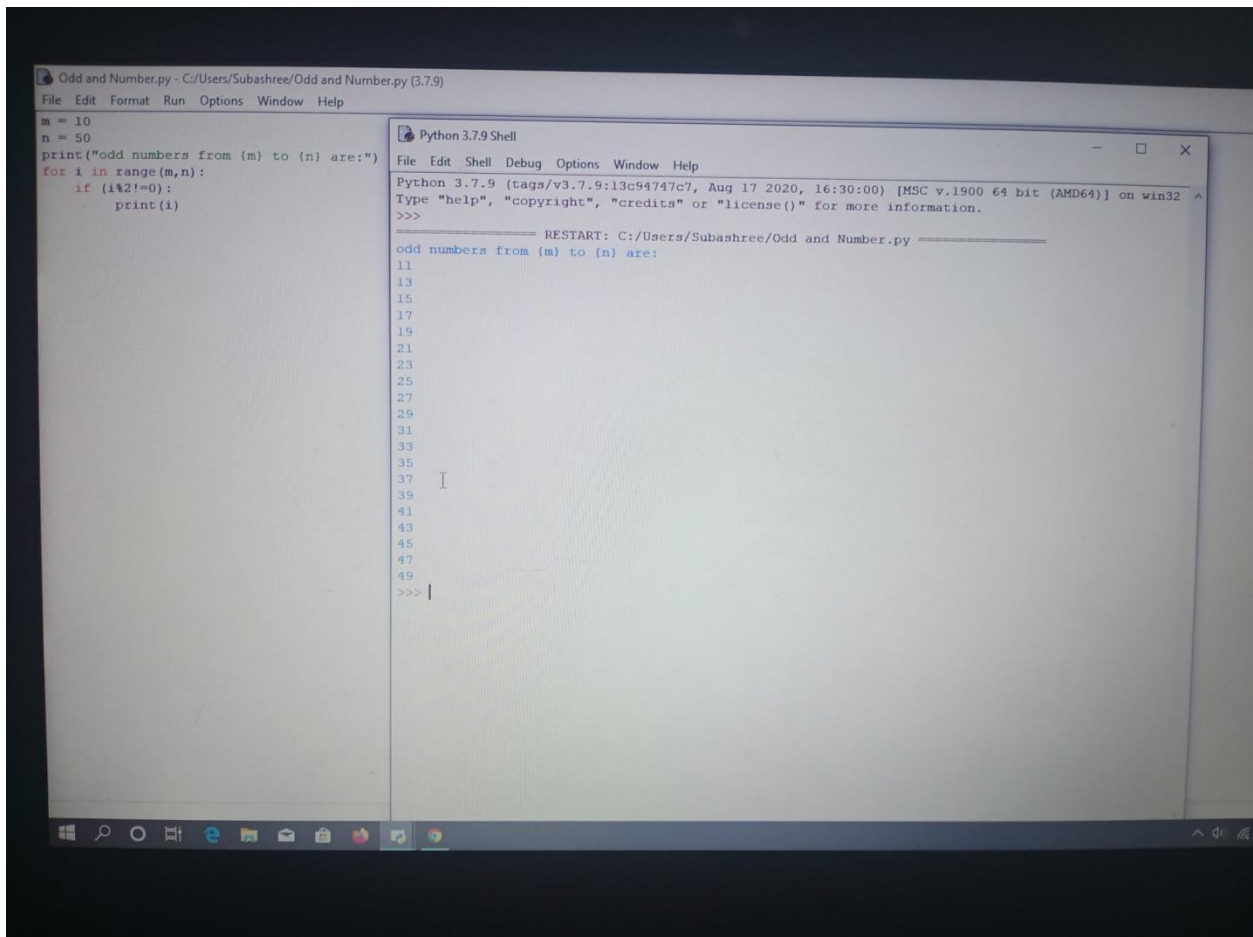
Enter the number23

number is prime

>>>

Ln 7 Col 4

2. Write a program to generate a odd numbers from m to n using while loop.



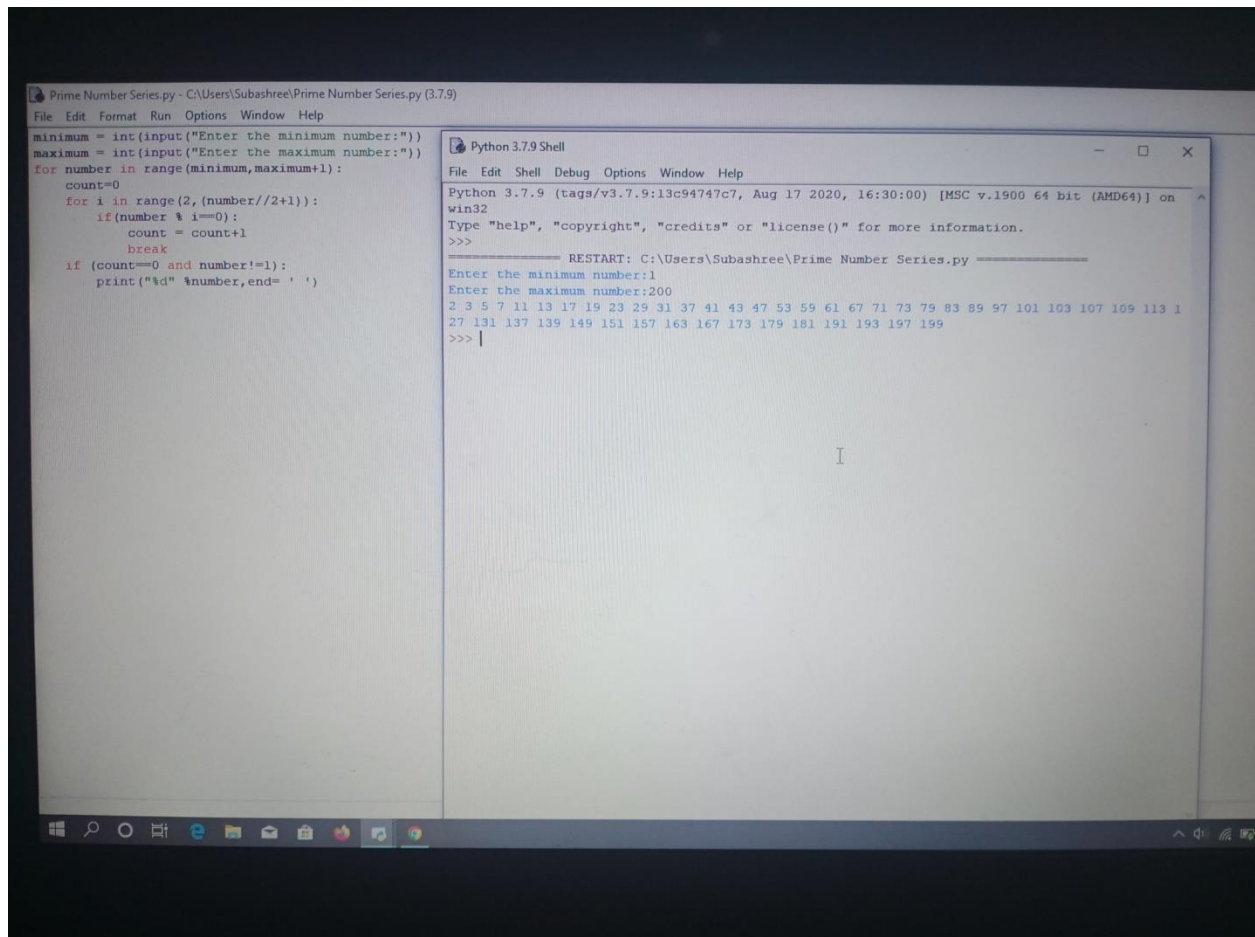
The image shows a screenshot of a Python IDE with two windows. The left window, titled 'Odd and Number.py - C:/Users/Subashree/Odd and Number.py (3.7.9)', contains the following code:

```
m = 10
n = 50
print("odd numbers from (m) to (n) are:")
for i in range(m,n):
    if (i%2!=0):
        print(i)
```

The right window, titled 'Python 3.7.9 Shell', shows the output of the program after execution. It displays the prompt 'Python 3.7.9 (tags/v3.7.9:13c94747c7, Aug 17 2020, 16:30:00) [MSC v.1900 64 bit (AMD64)] on win32' followed by the prompt 'Type "help", "copyright", "credits" or "license()" for more information.' and the prompt 'RESTART: C:/Users/Subashree/Odd and Number.py'. The output of the program is displayed as follows:

```
odd numbers from (m) to (n) are:
11
13
15
17
19
21
23
25
27
29
31
33
35
37
39
41
43
45
47
49
>>> |
```

3. Write a Python program to display prime number series up to given number.



The image shows a screenshot of a Python IDE with two windows. The left window, titled 'Prime Number Series.py - C:\Users\Subashree\Prime Number Series.py (3.7.9)', contains the following Python code:

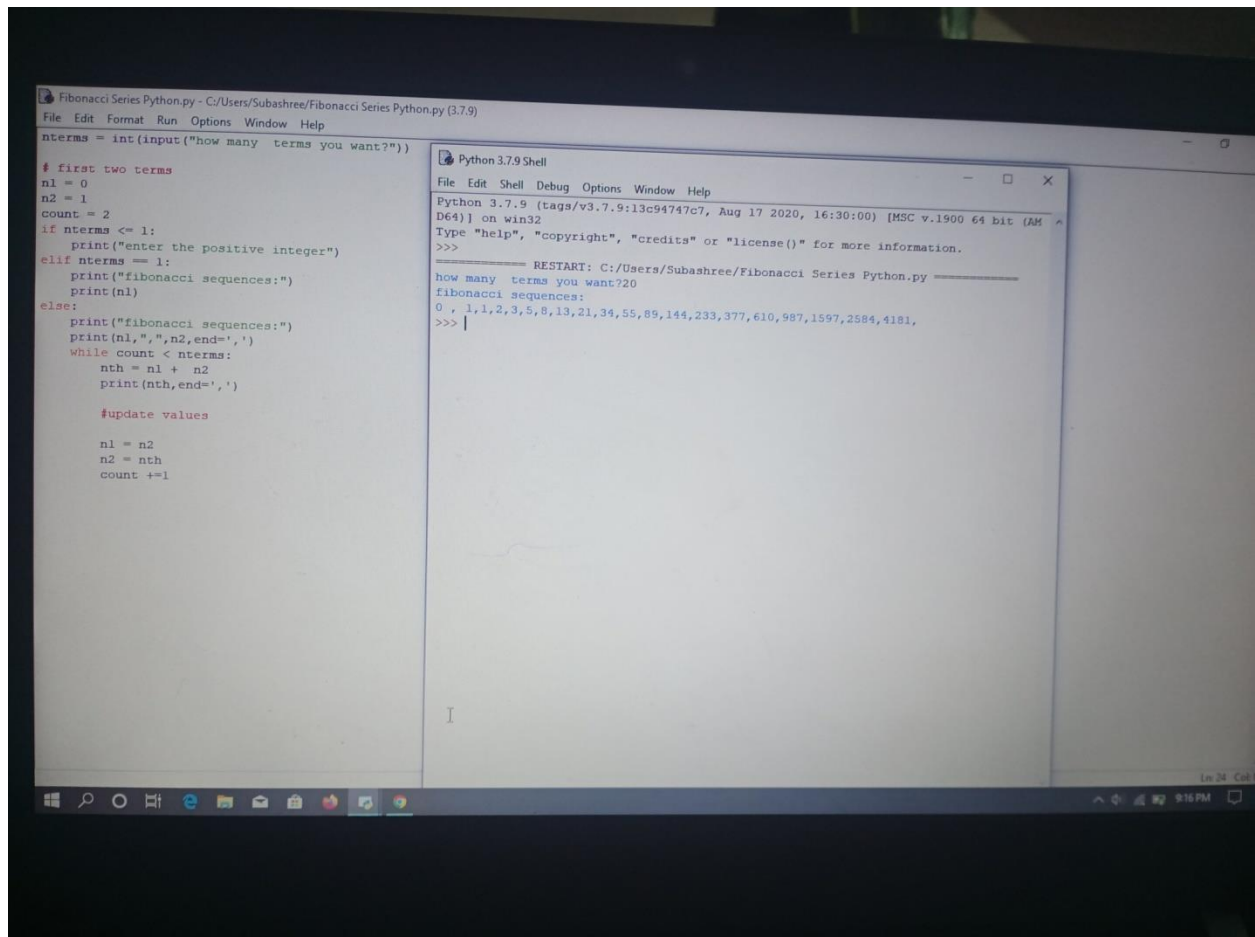
```
minimum = int(input("Enter the minimum number:"))
maximum = int(input("Enter the maximum number:"))
for number in range(minimum, maximum+1):
    count = 0
    for i in range(2, (number//2+1)):
        if (number % i == 0):
            count = count + 1
            break
    if (count == 0 and number != 1):
        print("%d" % number, end= ' ')
```

The right window, titled 'Python 3.7.9 Shell', shows the execution of the program. It displays the following text:

```
Python 3.7.9 (tags/v3.7.9:13c94747c7, Aug 17 2020, 16:30:00) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\Subashree\Prime Number Series.py =====
Enter the minimum number:1
Enter the maximum number:200
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97 101 103 107 109 113 1
27 131 137 139 149 151 157 163 167 173 179 181 191 193 197 199
>>> |
```

The output displays a list of prime numbers from 2 to 199, with a line break after the first row of numbers.

4. Write a Python program to generate Fibonacci series.



The image shows a screenshot of a Python 3.7.9 IDE. The left pane displays the source code for a program that generates a Fibonacci series. The code prompts the user for the number of terms, handles the first two terms (0 and 1), and then uses a while loop to calculate and print the subsequent terms. The right pane shows the Python 3.7.9 Shell with the program's execution output. The user entered 20 terms, and the program printed the sequence: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181.

```
Fibonacci Series Python.py - C:/Users/Subashree/Fibonacci Series Python.py (3.7.9)
File Edit Format Run Options Window Help
nterms = int(input("how many terms you want?"))

# first two terms
n1 = 0
n2 = 1
count = 2
if nterms <= 1:
    print("enter the positive integer")
elif nterms == 1:
    print("fibonacci sequences:")
    print(n1)
else:
    print("fibonacci sequences:")
    print(n1, ",", n2, end=', ')
    while count < nterms:
        nth = n1 + n2
        print(nth, end=', ')

        #update values
        n1 = n2
        n2 = nth
        count +=1

Python 3.7.9 Shell
File Edit Shell Debug Options Window Help
Python 3.7.9 (tags/v3.7.9:13c94747c7, Aug 17 2020, 16:30:00) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Subashree/Fibonacci Series Python.py =====
how many terms you want?20
fibonacci sequences:
0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181,
>>>
```