

IBM ASSIGNMENT 1

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

Program:

```
n = int(input("Enter a number: "))
flag = False
if n > 1:
    for i in range(2, n):
        if (n % i) == 0:
            flag = True
            break
if flag:
    print(n, "is not a prime number")
else:
    print(n, "is a prime number")
```



The screenshot shows a Python IDE with a dark theme. The editor window displays the code from the previous block. Below the editor, the 'TERMINAL' tab is active, showing the execution of the program. The command prompt shows the file path and the execution of the Python script. The output shows the program running twice: first with input 32, which is not a prime number, and then with input 31, which is a prime number.

```
First.py x
First.py > ...
1
2 num = int(input("Enter a number: "))
3 flag = False
4 if num > 1:
5     for i in range(2, num):
6         if (num % i) == 0:
7             flag = True
8             break
9 if flag:
10     print(num, "is not a prime number")
11 else:
12     print(num, "is a prime number")
13

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
PS C:\Users\Admin\Desktop\New folder> & C:/Users/Admin/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/Admin/Desktop/New folder/First.py"
Enter a number: 32
32 is not a prime number
PS C:\Users\Admin\Desktop\New folder> & C:/Users/Admin/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/Admin/Desktop/New folder/First.py"
Enter a number: 31
31 is a prime number
PS C:\Users\Admin\Desktop\New folder>

Ln 13, Col 1 Spaces: 4 UTF-8 CRLF Python 3.10.1 64-bit
```

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

Program:

```
m= int(input(" Please Enter the Maximum Value : "))
```

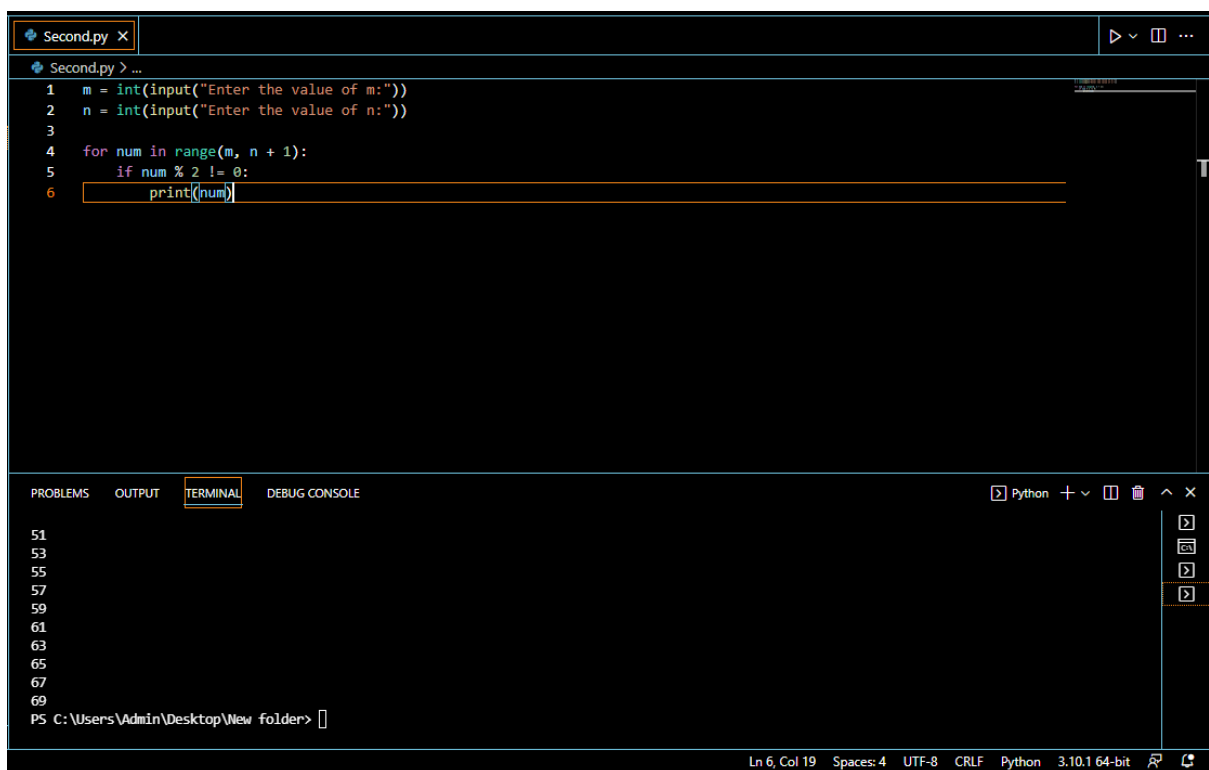
```
number = 1
```

```
while number <= maximum:
```

```
    if(number % 2 != 0):
```

```
        print("{0}".format(number))
```

```
    number = number + 1
```



The screenshot shows a Python IDE with a file named 'Second.py'. The code in the editor is as follows:

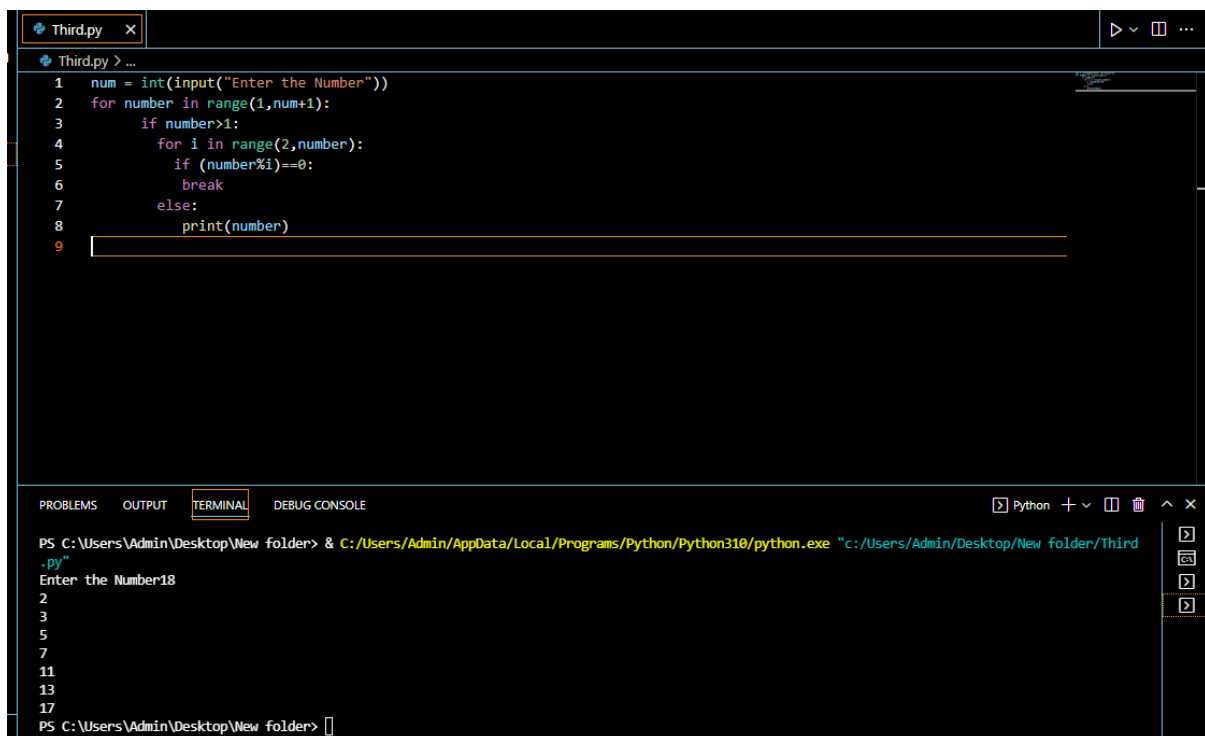
```
1 m = int(input("Enter the value of m:"))
2 n = int(input("Enter the value of n:"))
3
4 for num in range(m, n + 1):
5     if num % 2 != 0:
6         print(num)
```

The IDE interface includes a 'TERMINAL' tab at the bottom, which is currently empty. The status bar at the bottom right indicates 'Ln 6, Col 19', 'Spaces: 4', 'UTF-8', 'CRLF', 'Python', and '3.10.1 64-bit'.

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

Program:

```
num = int(input("Enter the Number"))
for number in range(1,num+1):
    if number>1:
        for i in range(2,number):
            if (number%i)==0:
                break
            else:
                print(number)
```



The screenshot displays a Python IDE with a file named 'Third.py'. The code in the editor is as follows:

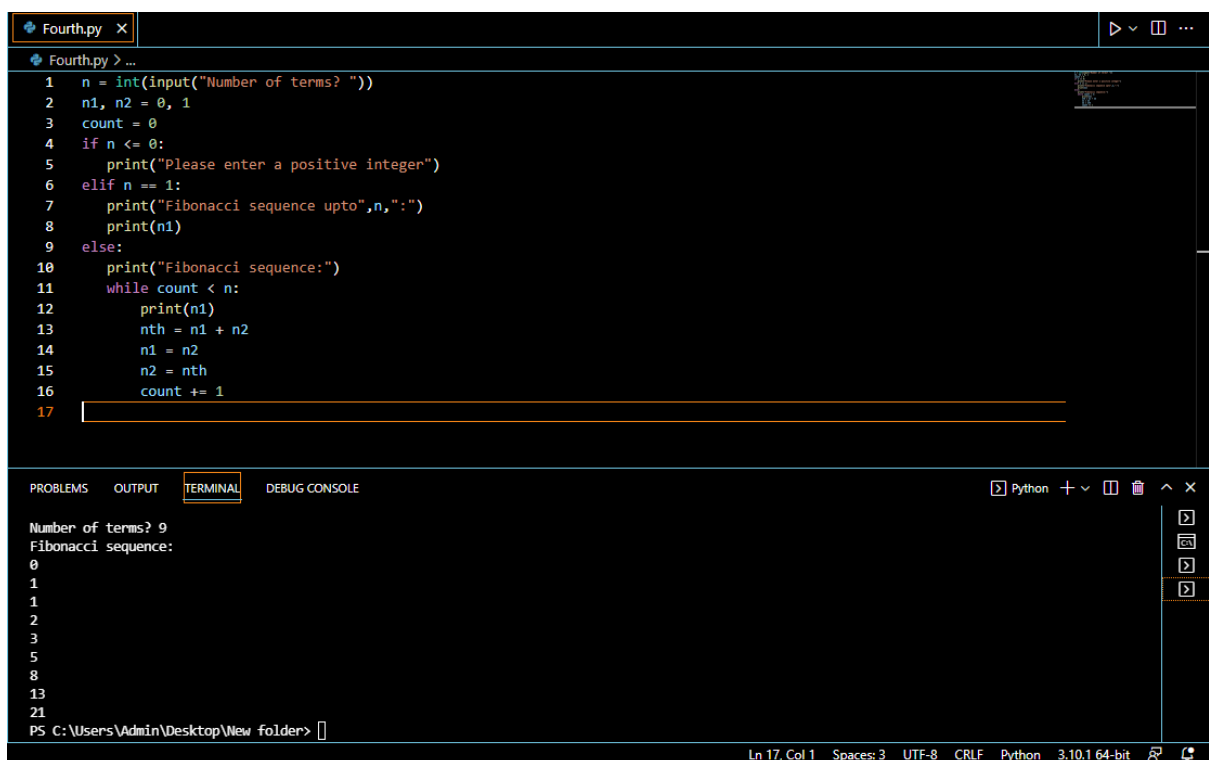
```
1 num = int(input("Enter the Number"))
2 for number in range(1,num+1):
3     if number>1:
4         for i in range(2,number):
5             if (number%i)==0:
6                 break
7             else:
8                 print(number)
9
```

The terminal window at the bottom shows the command prompt execution. The command is `PS C:\Users\Admin\Desktop\New folder> & C:/Users/Admin/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/Admin/Desktop/New folder/Third.py"`. The output shows the program running and printing the prime numbers 2, 3, 5, 7, 11, 13, and 17, which are the prime numbers up to 18.

4. Write a python program to generate fibonacci series

Program:

```
nterms = int(input("Number of terms? "))
n1, n2 = 0, 1
count = 0
if nterms <= 0:
    print("Please enter a positive integer")
elif nterms == 1:
    print("Fibonacci sequence upto", nterms, ":")
    print(n1)
else:
    print("Fibonacci sequence:")
    while count < nterms:
        print(n1)
        nth = n1 + n2
        n1 = n2
        n2 = nth
        count += 1
```



The screenshot shows a Python IDE with a file named 'Fourth.py'. The code in the editor is as follows:

```
1 n = int(input("Number of terms? "))
2 n1, n2 = 0, 1
3 count = 0
4 if n <= 0:
5     print("Please enter a positive integer")
6 elif n == 1:
7     print("Fibonacci sequence upto", n, ":")
8     print(n1)
9 else:
10    print("Fibonacci sequence:")
11    while count < n:
12        print(n1)
13        nth = n1 + n2
14        n1 = n2
15        n2 = nth
16        count += 1
17
```

The terminal output shows the program running with 9 terms entered:

```
Number of terms? 9
Fibonacci sequence:
0
1
1
2
3
5
8
13
21
```

The status bar at the bottom indicates the current position is Line 17, Column 1, with 3 spaces, UTF-8 encoding, CRLF line endings, Python 3.10.1 64-bit.