

IBM ASSIGNMENT 1:

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

PROGRAM

```
num=int(input("enter a number:"))

for x in range(2,num):

    if num%x==0:

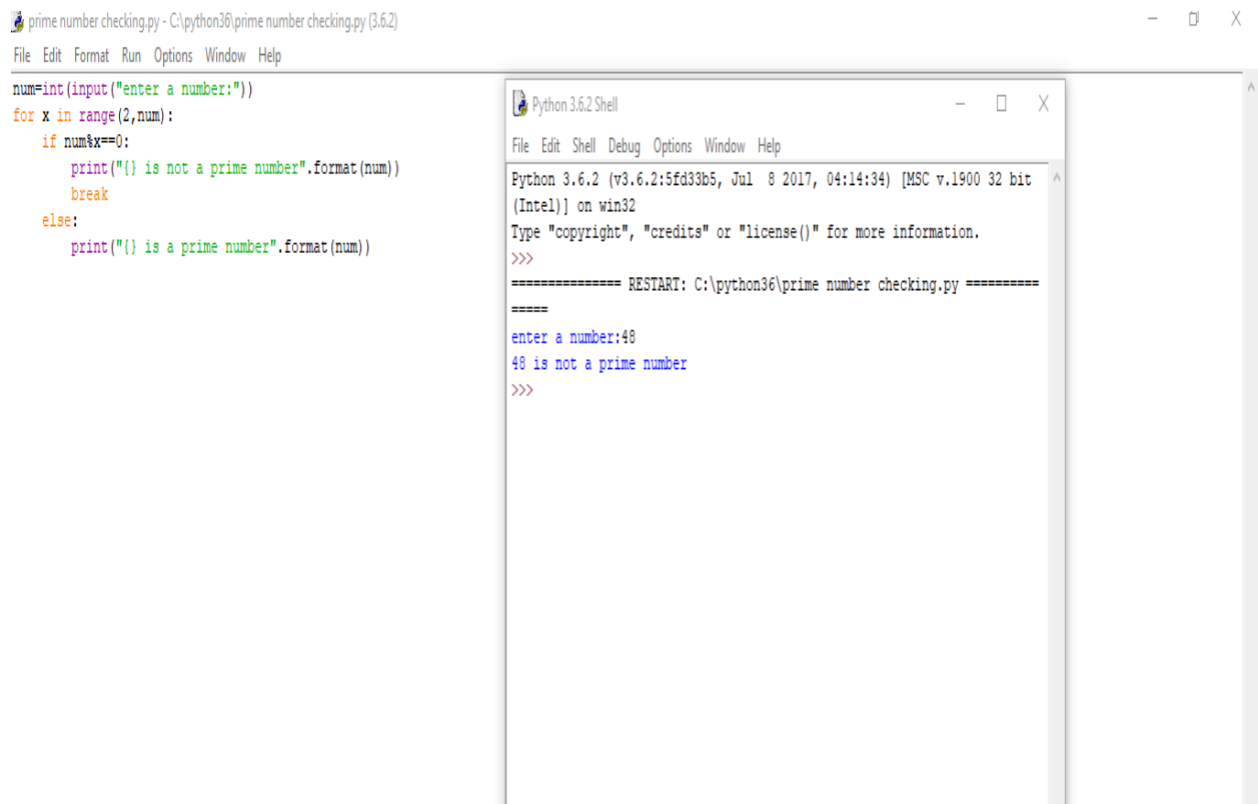
        print("{} is not a prime number".format(num))

        break

    else:

        print("{} is a prime number".format(num))
```

OUTPUT



The screenshot displays a Python IDE with two windows. The left window, titled 'prime number checking.py - C:\python36\prime number checking.py (3.6.2)', shows the source code for a prime number checker. The right window, titled 'Python 3.6.2 Shell', shows the execution output. The code in the left window is as follows:

```
num=int(input("enter a number:"))
for x in range(2,num):
    if num%x==0:
        print("{} is not a prime number".format(num))
        break
    else:
        print("{} is a prime number".format(num))
```

The output in the right window shows the program running on Python 3.6.2. It prompts the user to enter a number, and when 48 is entered, it outputs '48 is not a prime number'.

```
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\python36\prime number checking.py =====
>>>
enter a number:48
48 is not a prime number
>>>
```

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

PROGRAM

```
first=int(input("enter first number"))
```

```
last=int(input("enter last number"))
```

```
for n in range(first,last+1):
```

```
    if n > 1:
```

```
        for i in range(2,n):
```

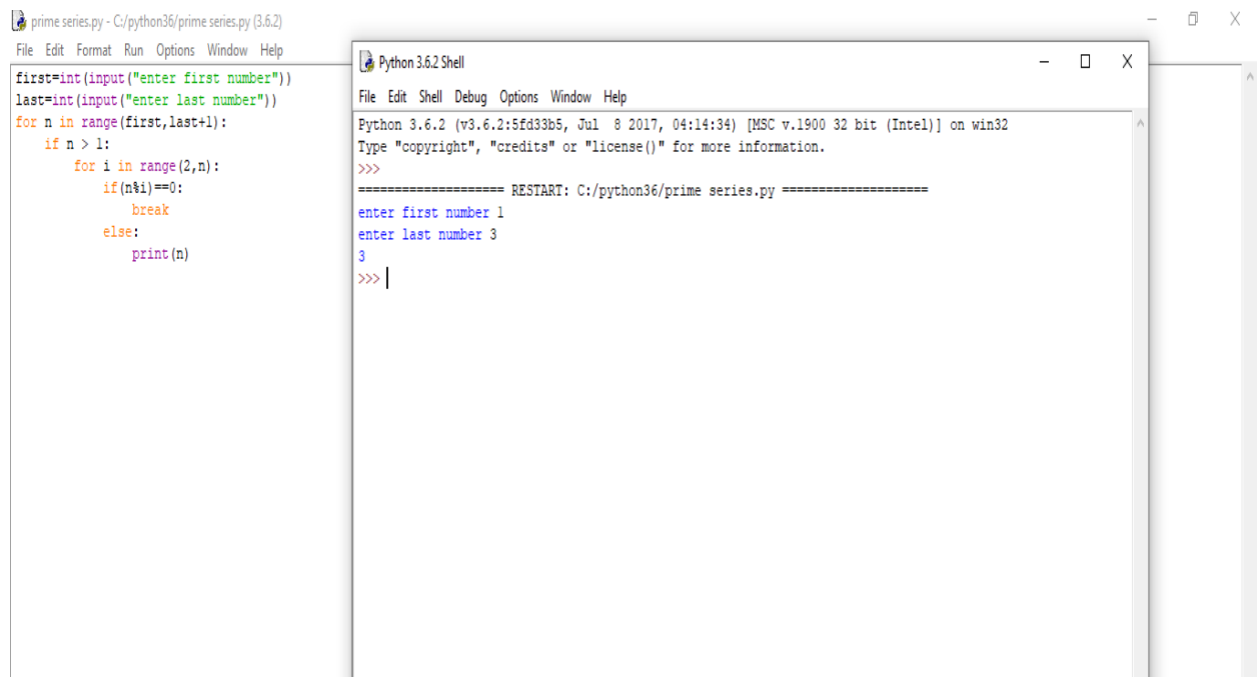
```
            if(n%i)==0:
```

```
                break
```

```
            else:
```

```
                print(n)
```

OUTPUT



The screenshot displays two windows from a Python IDE. The left window, titled 'prime series.py - C:/python36/prime series.py (3.6.2)', contains the Python code for generating prime numbers. The code prompts the user for a first and last number, then iterates through the range from first to last+1. For each number n, it checks if it is greater than 1. If so, it enters a loop from 2 to n. If n is divisible by any number in this range (n%i == 0), it breaks the loop. If the loop completes without breaking, n is a prime number and is printed. The right window, titled 'Python 3.6.2 Shell', shows the execution of the program. It displays the Python version and system information. After a restart, it shows the user inputting '1' for the first number and '3' for the last number. The output of the program is the number '3', which is the only prime number in the range [1, 3].

```
prime series.py - C:/python36/prime series.py (3.6.2)
File Edit Format Run Options Window Help
first=int(input("enter first number"))
last=int(input("enter last number"))
for n in range(first,last+1):
    if n > 1:
        for i in range(2,n):
            if(n%i)==0:
                break
            else:
                print(n)

Python 3.6.2 Shell
File Edit Shell Debug Options Window Help
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/python36/prime series.py =====
enter first number 1
enter last number 3
3
>>> |
```

3. Write a python program to generate Fibonacci series

PROGRAM

```
n = int(input("\nPlease Enter the Range : "))
```

```
i = 0
```

```
First_Value = 0
```

```
Second_Value = 1
```

```
while(i < n):
```

```
if(i <= 1):
```

```
Next = i
```

```
else:
```

```
Next = First_Value + Second_Value
```

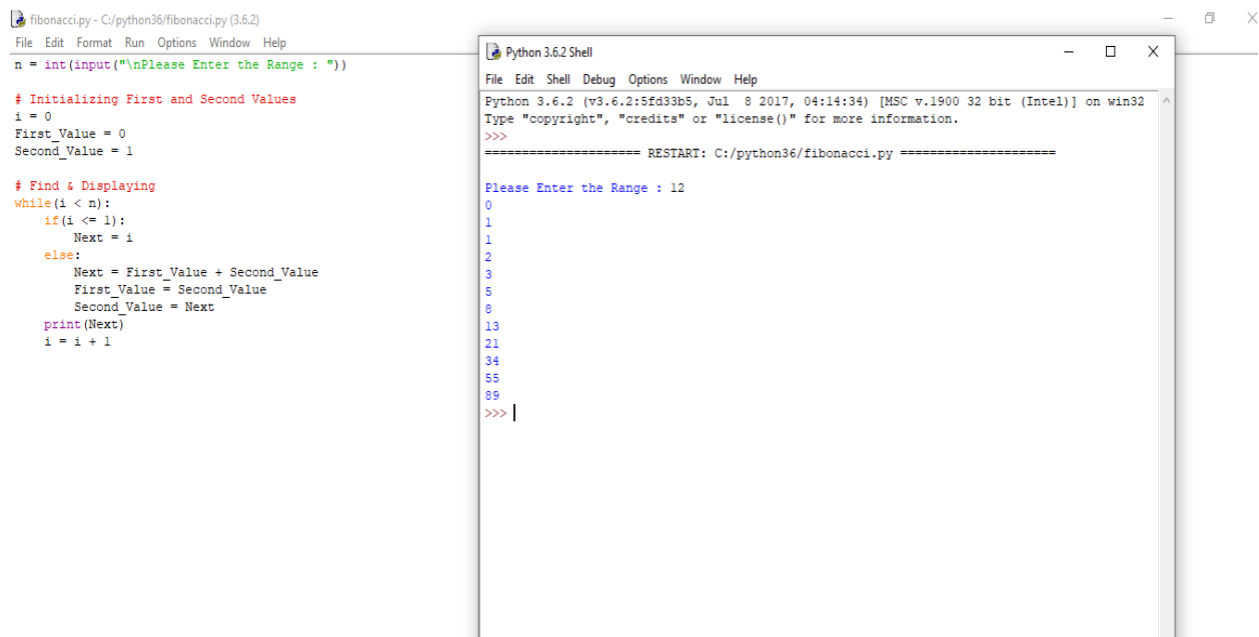
```
First_Value = Second_Value
```

```
Second_Value = Next
```

```
print(Next)
```

```
i = i + 1
```

OUTPUT



```
fibonacci.py - C:/python36/fibonacci.py (3.6.2)
File Edit Format Run Options Window Help
n = int(input("\nPlease Enter the Range : "))

# Initializing First and Second Values
i = 0
First_Value = 0
Second_Value = 1

# Find & Displaying
while(i < n):
    if(i <= 1):
        Next = i
    else:
        Next = First_Value + Second_Value
        First_Value = Second_Value
        Second_Value = Next
    print(Next)
    i = i + 1
```

```
Python 3.6.2 Shell
File Edit Shell Debug Options Window Help
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/python36/fibonacci.py =====
Please Enter the Range : 12
0
1
1
2
3
5
8
13
21
34
55
89
>>> |
```

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

PROGRAM

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

number = 1

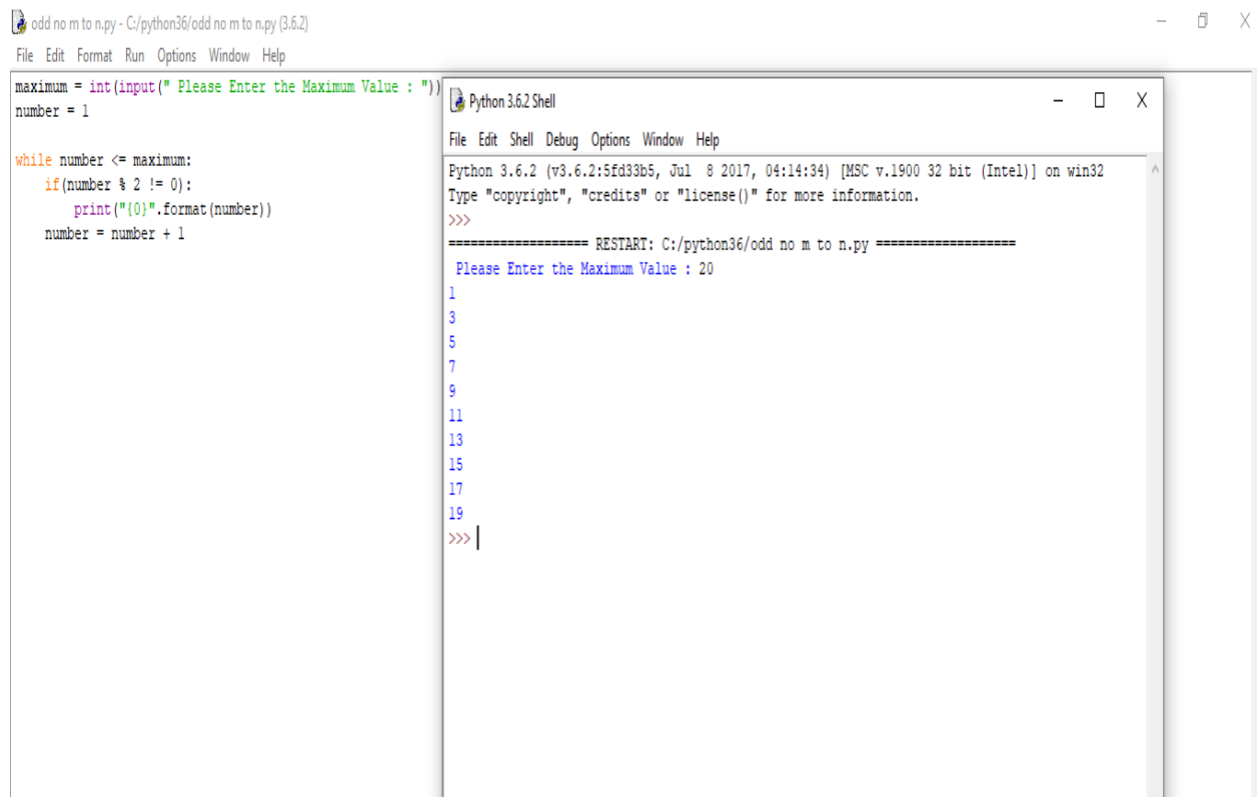
while number <= maximum:

    if(number % 2 != 0):

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

    number = number + 1
```

OUTPUT



The screenshot displays a Python IDE with two windows. The left window, titled 'odd no m to n.py - C:/python36/odd no m to n.py (3.6.2)', contains the following code:

```
maximum = int(input(" Please Enter the Maximum Value : "))
number = 1

while number <= maximum:
    if(number % 2 != 0):
        print("{0}".format(number))
    number = number + 1
```

The right window, titled 'Python 3.6.2 Shell', shows the execution output:

```
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/python36/odd no m to n.py =====
Please Enter the Maximum Value : 20
1
3
5
7
9
11
13
15
17
19
>>> |
```