

ASSIGNMENT – 1

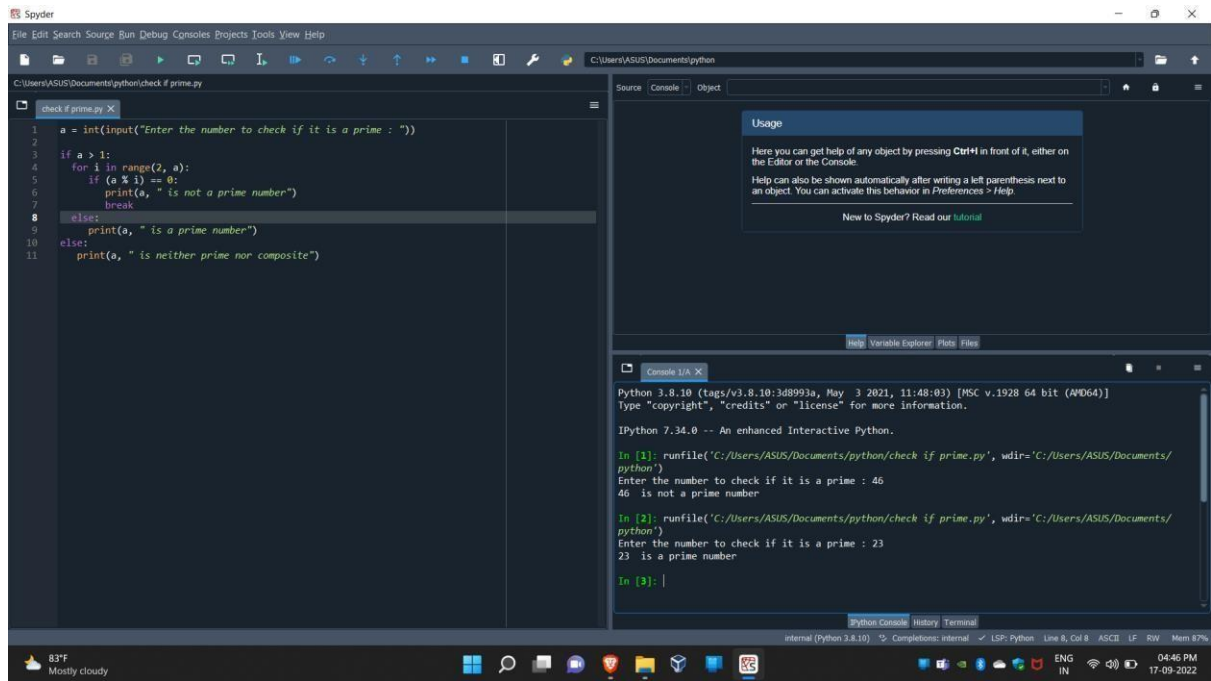
TEAM ID	PNT2022TMID33820
PROJECT NAME	PERSONAL EXPENSE TRACKER APPLICATION
NAME	Jeevanandham G
ROLL NO	950819104018

1.Check if prime or not:

Program:

```
a = int(input("Enter the number to check if it is a prime : "))
if a > 1:
    for i in range(2, a):
        if (a % i) == 0:
            print(a, " is not a prime number")
            break
    else:
        print(a, " is a prime number")
else:
    print(a, " is neither prime nor composite")
```

Output:



2.Generate odd number from m to n using while loop:

Program:

```
print("Finding odd numbers in a given range ... ")

m = int(input("From : "))

n = int(input("To :"))

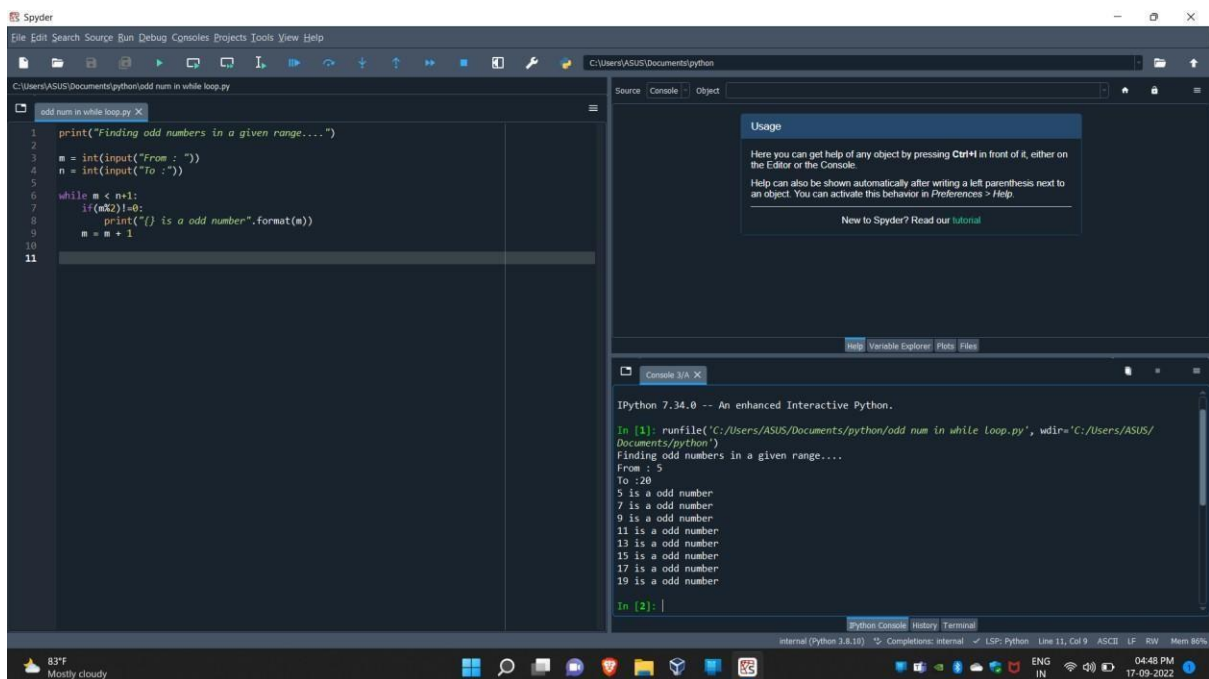
while m < n+1:

    if(m%2)!=0:

        print("{} is a odd number".format(m))

    m = m + 1
```

Output:



3.Display prime number series upto given number:

Program:

```
a = int(input("Enter the lower bound: "))
b = int(input("Enter the upper bound: "))

for i in range(a,b+1):

    if i > 1:

        for j in range(2, i):

            if (i % j) == 0:

                break

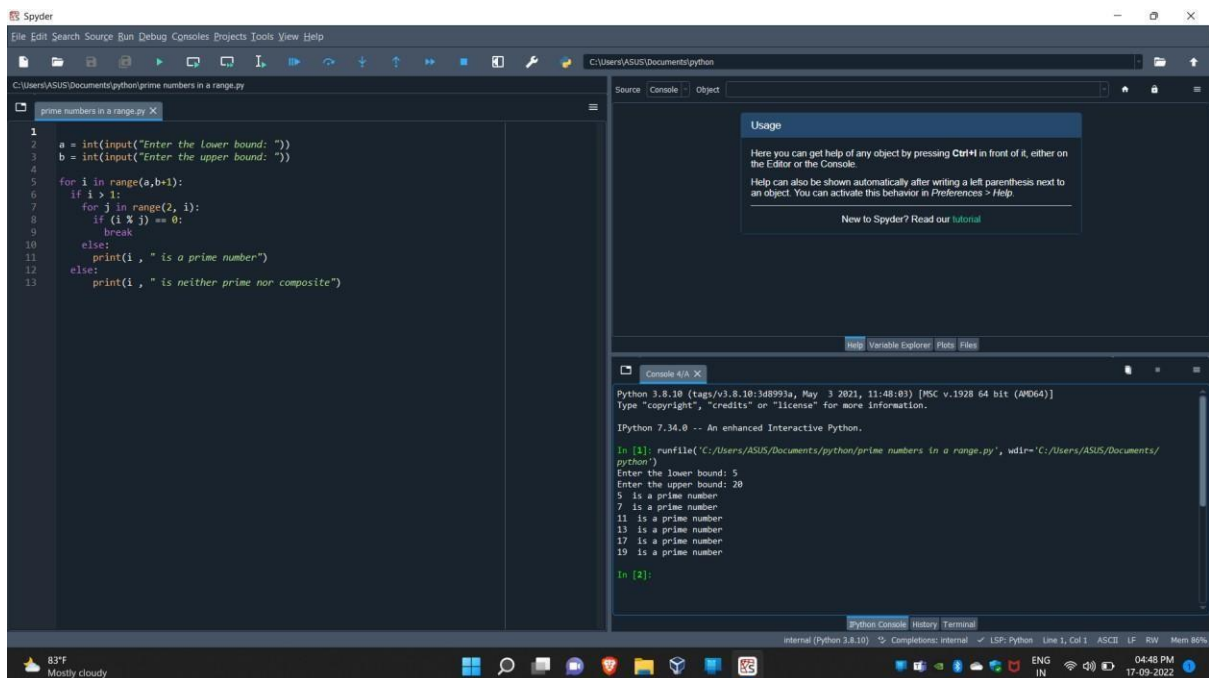
        else:

            print(i , " is a prime number")

    else:

        print(i , " is neither prime nor composite")
```

Output:



The screenshot displays the Spyder Python IDE interface. The main editor window on the left contains a Python script named 'prime numbers in a range.py'. The script prompts the user for a lower bound and an upper bound, then iterates through the range from the lower bound to the upper bound (inclusive). For each number, it checks if it is a prime by testing divisibility from 2 up to the number itself. If a divisor is found, it breaks the inner loop. If no divisor is found (the inner loop completes), it prints that the number is a prime. Otherwise, it prints that the number is neither prime nor composite.

The console window on the right shows the output of the program. It displays the prompts 'Enter the lower bound: 5' and 'Enter the upper bound: 20', followed by the printed results: '5 is a prime number', '7 is a prime number', '11 is a prime number', '13 is a prime number', '17 is a prime number', and '19 is a prime number'.

```
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\ASUS\Documents\python\prime numbers in a range.py
1 a = int(input("Enter the lower bound: "))
2 b = int(input("Enter the upper bound: "))
3
4
5 for i in range(a,b+1):
6     if i > 1:
7         for j in range(2, i):
8             if (i % j) == 0:
9                 break
10        else:
11            print(i , " is a prime number")
12    else:
13        print(i , " is neither prime nor composite")

Source Console Object
Usage
Here you can get help of any object by pressing Ctrl+H in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.
New to Spyder? Read our tutorial

Help Variable Explorer Plots Files
Console 4/4 X
Python 3.8.10 (tags/v3.8.10:3d8993a, May 3 2021, 11:48:03) [MSC v.1928 64 bit (AMD64)]
Type "copyright", "credits" or "license()" for more information.
IPython 7.34.0 -- An enhanced Interactive Python.
In [1]: runfile('C:/Users/ASUS/Documents/python/prime numbers in a range.py', wdir='C:/Users/ASUS/Documents/python')
Enter the lower bound: 5
Enter the upper bound: 20
5 is a prime number
7 is a prime number
11 is a prime number
13 is a prime number
17 is a prime number
19 is a prime number
In [2]:
```

4.Generate Fibonacci Series:

Program:

```
a = 0
```

```
b = 1
```

```
n = int(input("Enter the range of fibonacci numbers you wish to find : "))
```

```
print(a)
```

```
print(b)
```

```
for i in range(0,n-2):
```

```
    fib = a + b
```

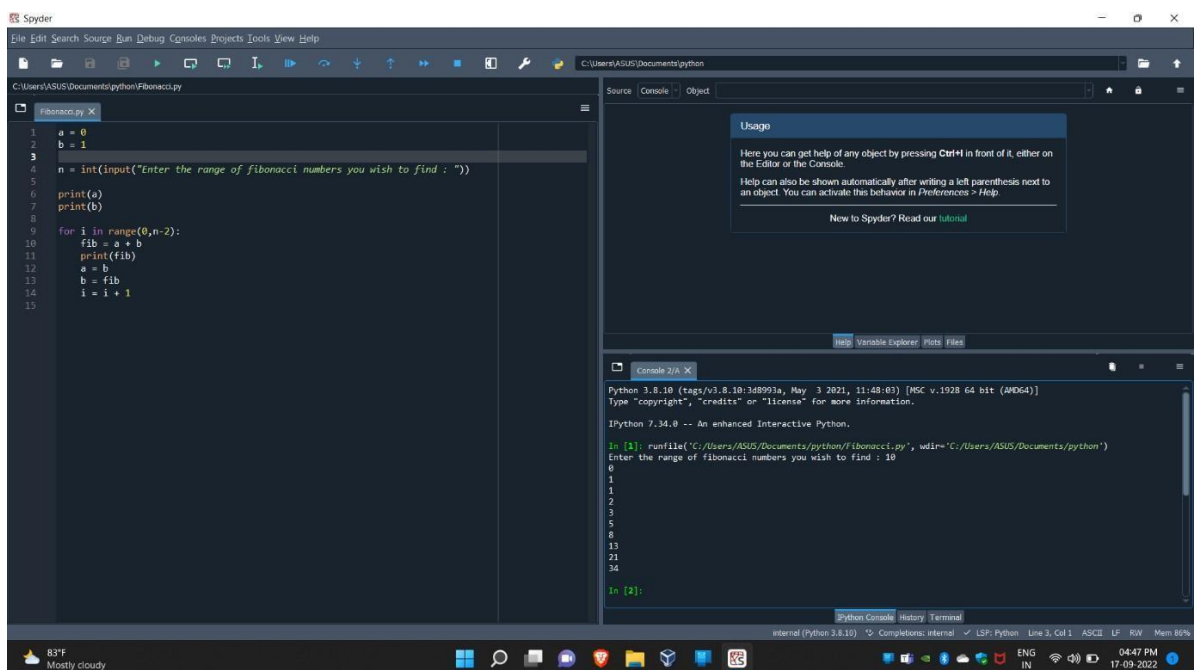
```
    print(fib)
```

```
    a = b
```

```
    b = fib
```

```
    i = i + 1
```

Output:



The screenshot displays the Spyder Python IDE interface. The left pane shows the source code for a file named 'Fibonacci.py'. The code defines variables 'a' and 'b' as 0 and 1, prompts the user for a range 'n', and prints the first two Fibonacci numbers. A loop then generates and prints the series up to 'n-2'. The right pane features a 'Usage' tip and a 'Python Console' showing the execution of the script. The console output displays the sequence of Fibonacci numbers: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34. The bottom status bar indicates the system is at 83°F, mostly cloudy, with a time of 04:47 PM on 17-09-2022.

```
1 a = 0
2 b = 1
3
4 n = int(input("Enter the range of fibonacci numbers you wish to find : "))
5
6 print(a)
7 print(b)
8
9 for i in range(0,n-2):
10     fib = a + b
11     print(fib)
12     a = b
13     b = fib
14     i = i + 1
15
```

Python 3.8.10 (tags/v3.8.10:348923a, May 3 2021, 11:45:03) [AMD64]
Type "copyright", "credits" or "license()" for more information.
IPython 7.34.0 -- An enhanced Interactive Python.
In [1]: runfile('C:/Users/ASUS/Documents/python/Fibonacci.py', wdir='C:/Users/ASUS/Documents/python')
Enter the range of fibonacci numbers you wish to find : 10
0
1
1
2
3
5
8
13
21
34
In [2]: