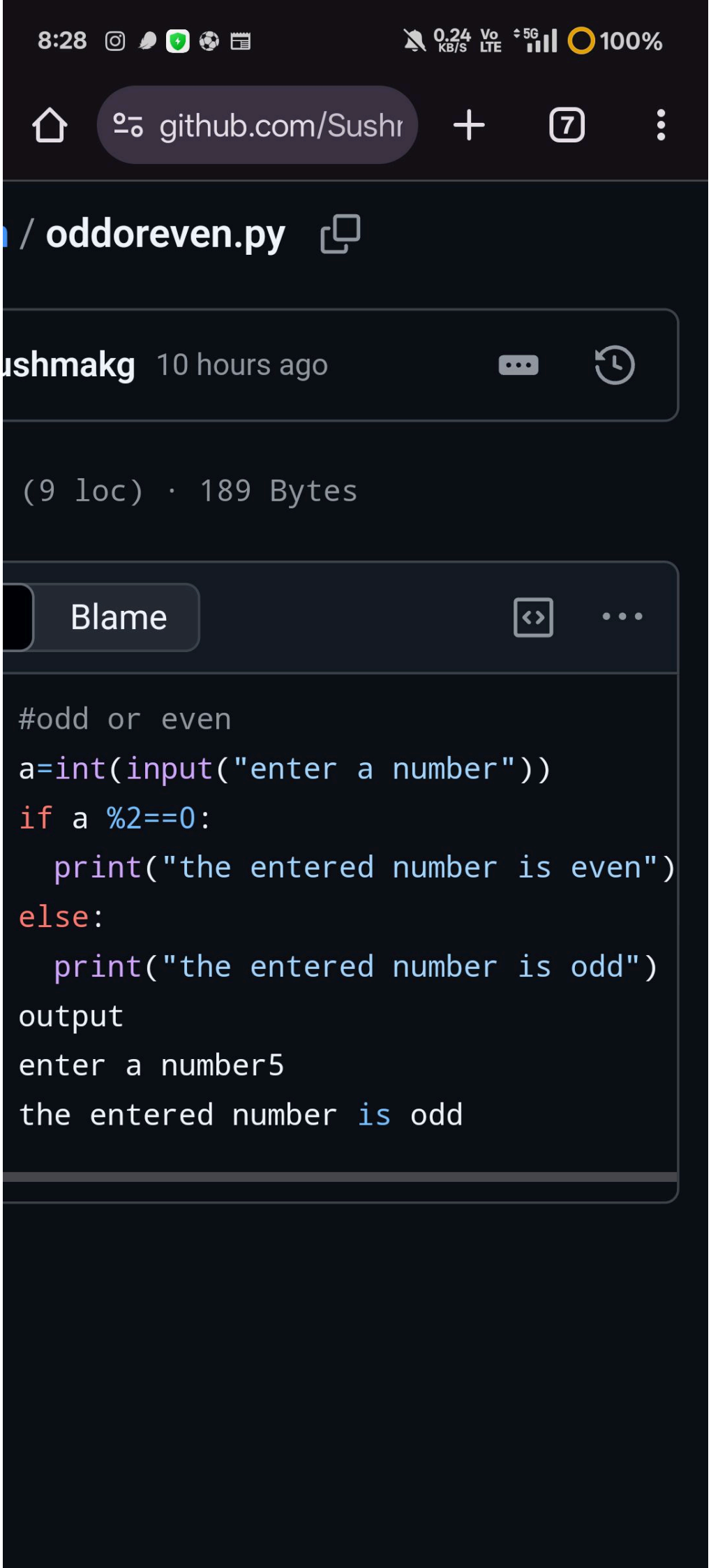
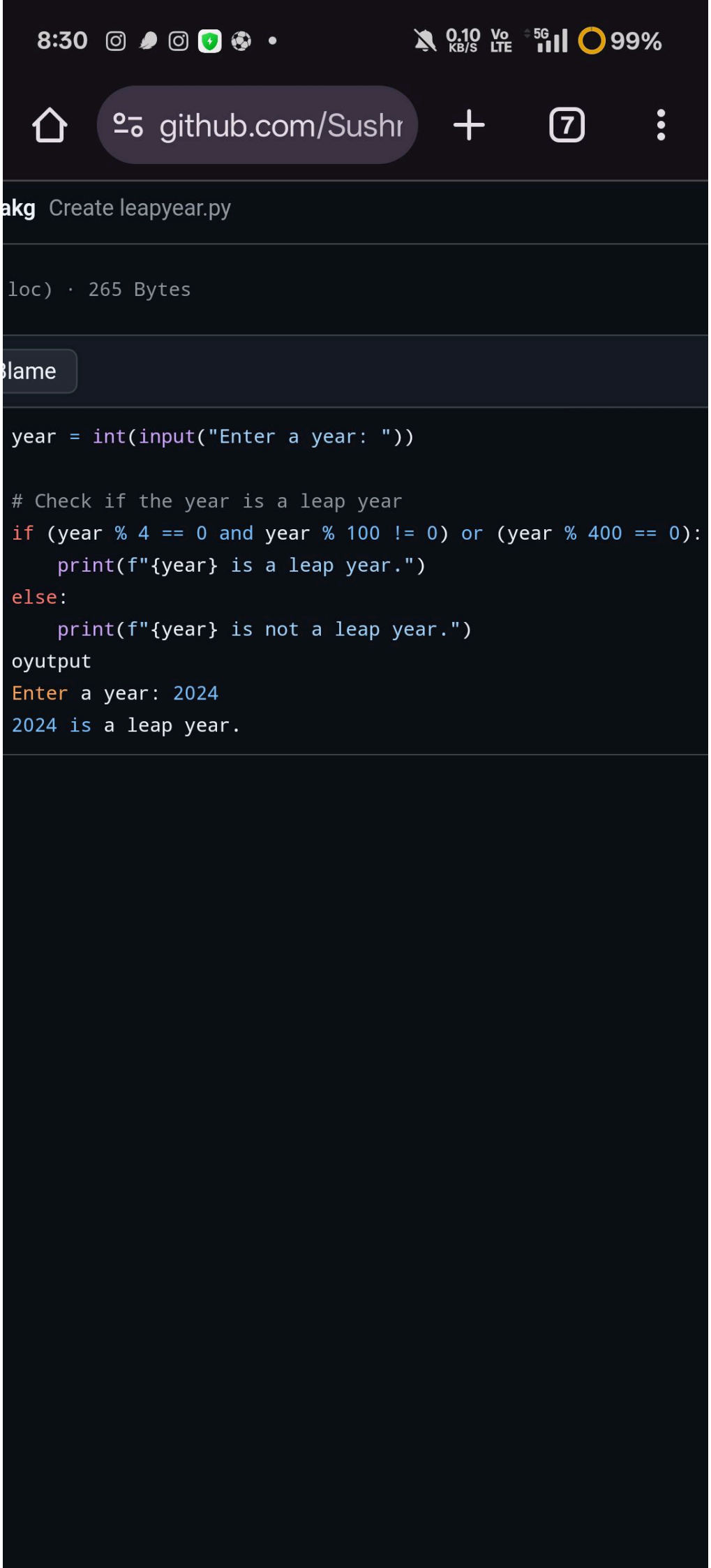
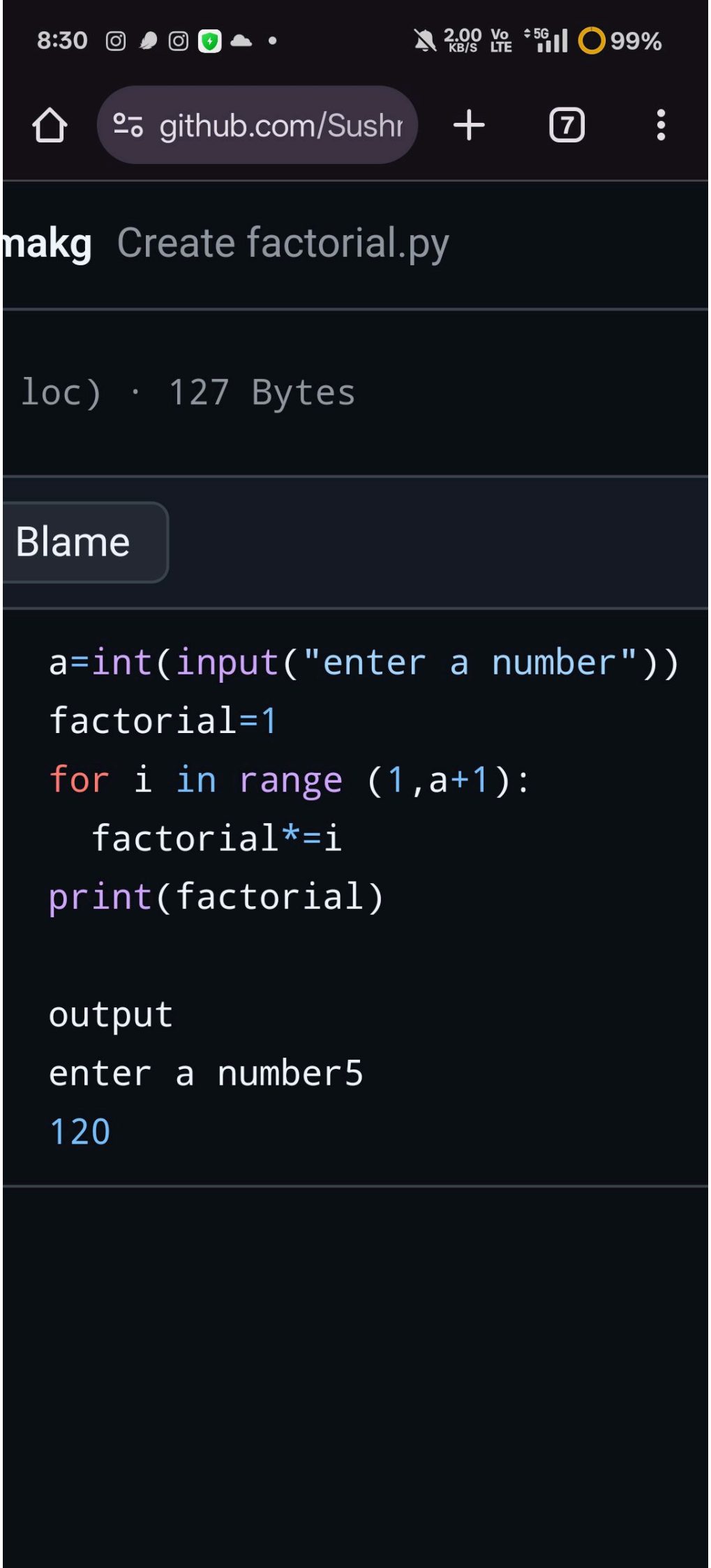
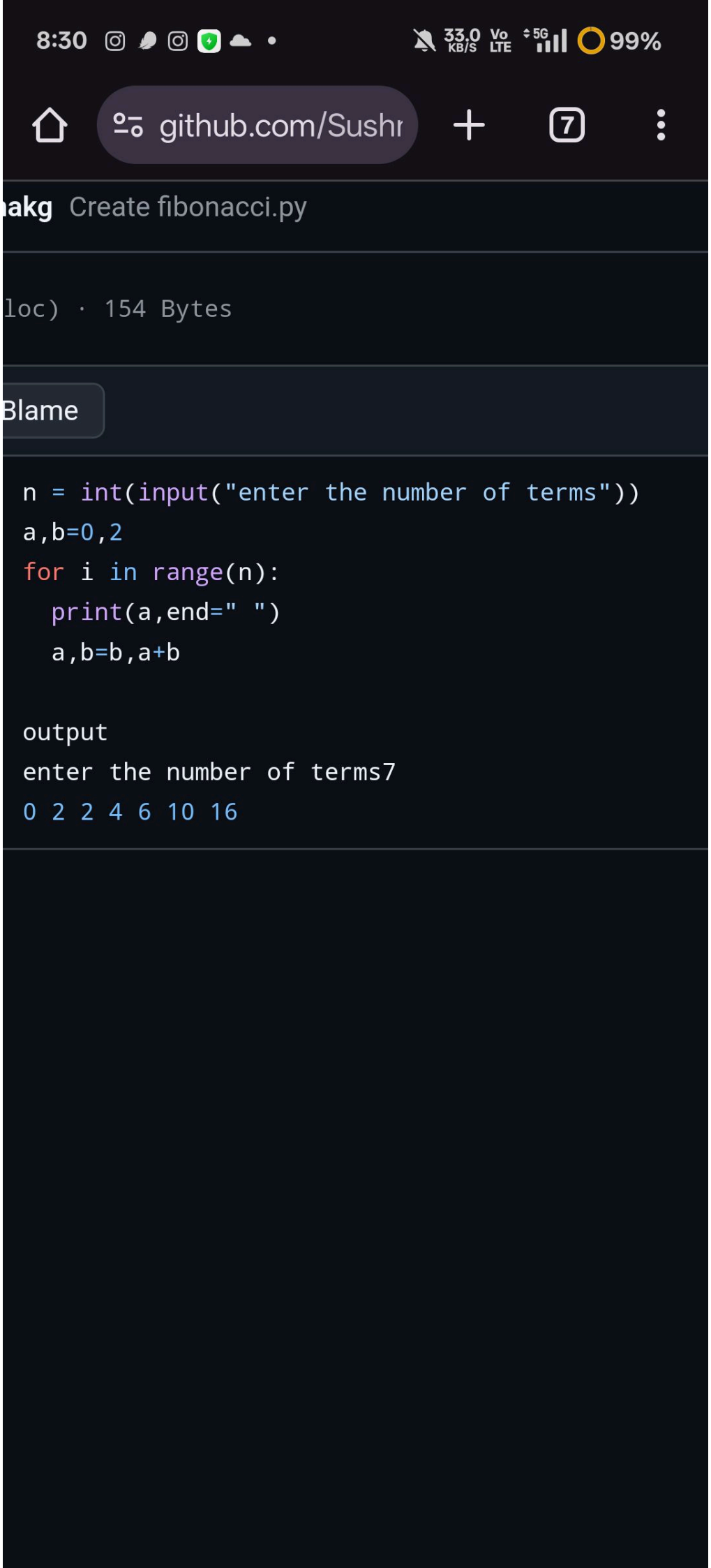
```
a=int(input("enter a number"))
if a>0:
  print("a is positive")
elif a<0:
  print("a is negative")
else:
  print("a is zero")

output
enter a number9
a is positive
```

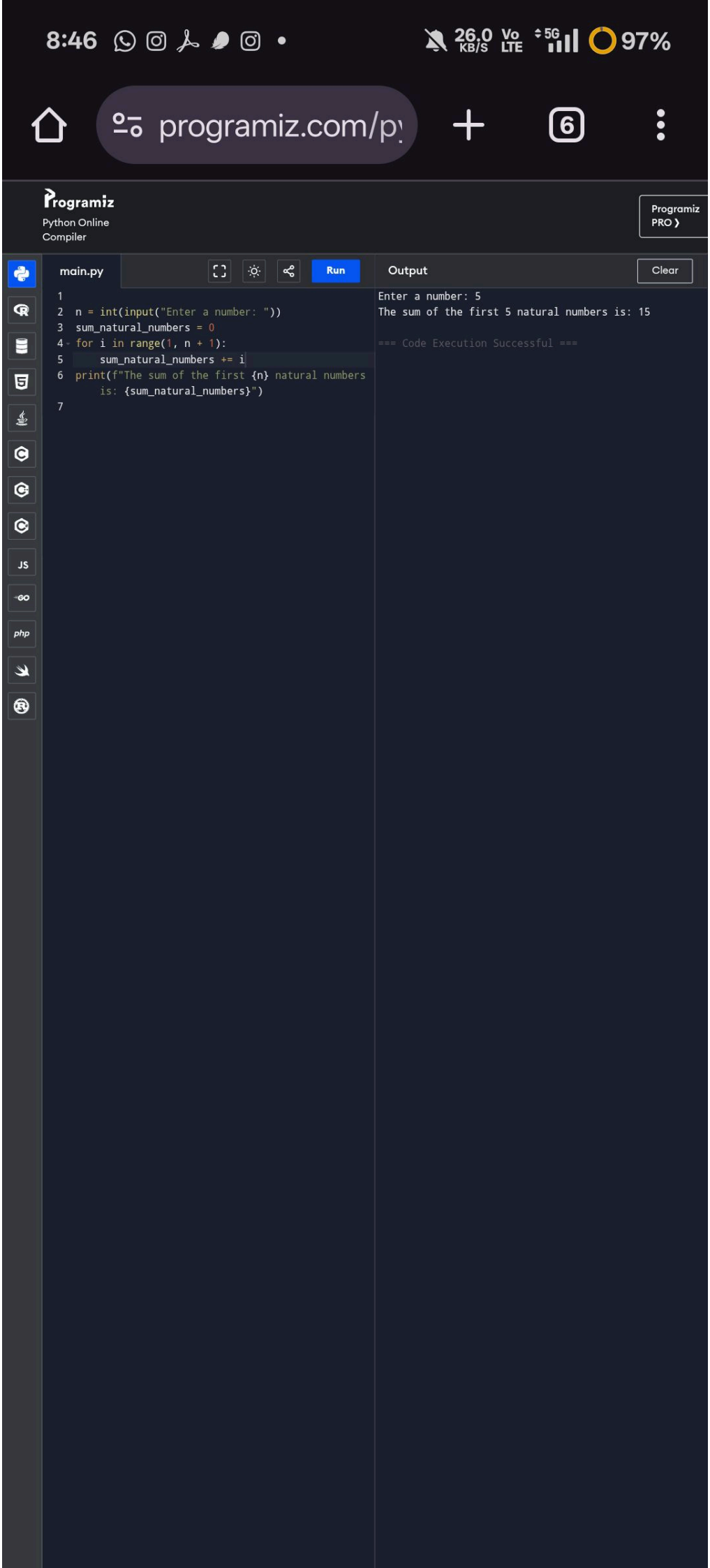

8:28
0.24 KB/S
Vo LTE
5G
100%
github.com/Sushr
7
/ oddoreven.py
ushmakg 10 hours ago
(9 loc) · 189 Bytes
Blame
#odd or even
a=int(input("enter a number"))
if a %2==0:
print("the entered number is even")
else:
print("the entered number is odd")
output
enter a number5
the entered number is odd

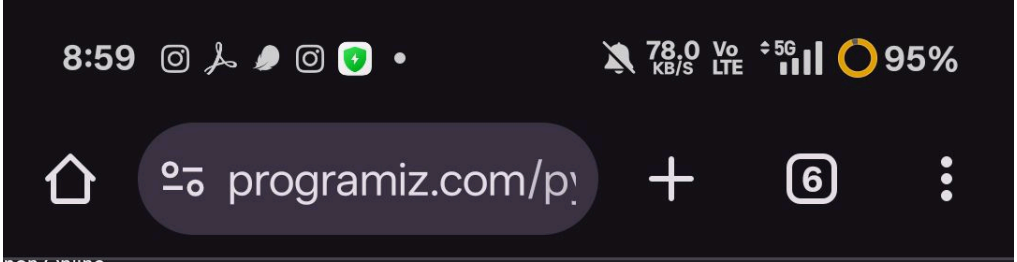
```
  # Taking input from the user
a = float(input("Enter first number: "))
b = float(input("Enter second number: "))
c = float(input("Enter third number: "))

# Using if-elif-else to find the largest
if a >= b and a >= c:
    print(f"The largest number is {a}")
elif b >= a and b >= c:
    print(f"The largest number is {b}")
else:
    print(f"The largest number is {c}")

output
Enter first number: 5
Enter second number: 8
Enter third number: 3
The largest number is 8
```


8:30
0.10 KB/S
Vo LTE
5G
99%
github.com/Sushr
7
akg Create leapyear.py
loc) · 265 Bytes
Blame
year = int(input("Enter a year: "))
# Check if the year is a leap year
if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):
print(f"{year} is a leap year.")
else:
print(f"{year} is not a leap year.")
oyutput
Enter a year: 2024
2024 is a leap year.


8:30
2.00 KB/S
99%
github.com/Sushr
7
makg Create factorial.py
loc) · 127 Bytes
Blame
a=int(input("enter a number"))
factorial=1
for i in range (1,a+1):
factorial*=i
print(factorial)
output
enter a number5
120


8:30
33.0 KB/S
Vo LTE
5G
99%
github.com/Sushr
7
akg Create fibonacci.py
loc) · 154 Bytes
Blame
n = int(input("enter the number of terms"))
a,b=0,2
for i in range(n):
print(a,end=" ")
a,b=b,a+b
output
enter the number of terms7
0 2 2 4 6 10 16


8:46
26.0 KB/S
Vo LTE
5G
97%
programiz.com/p
Programiz
Python Online Compiler
Programiz PRO
main.py
Run
Output
Clear
1
2 n = int(input("Enter a number: "))
3 sum_natural_numbers = 0
4 for i in range(1, n + 1):
5 sum_natural_numbers += i
6 print(f"The sum of the first {n} natural numbers is: {sum_natural_numbers}")
7
Enter a number: 5
The sum of the first 5 natural numbers is: 15
=== Code Execution Successful ===
JS
GO
php



hon Online
mpiler

main.py

Run

Output

```
1
2  number = int(input("Enter an integer: "))
3  for i in range(1, 11):
4      print(f"{number} x {i} = {number * i}")
5
```

```
Enter an integer: 10
10 x 1 = 10
10 x 2 = 20
10 x 3 = 30
10 x 4 = 40
10 x 5 = 50
10 x 6 = 60
10 x 7 = 70
10 x 8 = 80
10 x 9 = 90
10 x 10 = 100

=== Code Execution Successful ==
```

ain.py

Run

```
n=int(input("enter the value of n"))
if n<2:
    print("{n} is not a prime number")
else:
    for i in range (2,n):
        if n%i==0:
            print("{n} is not a prime number")
            break
    else:
        print("{n} is a prime number")
```

Output

```
enter the value of n7
{n} is a prime number

=== Code Execution Succes
```


9:20
0.28 KB/S
Vo LTE
5G
92%
programiz.com/p
on Online
piler
main.py
Run
Output
number = int(input("Enter an integer: "))
reversed_number = int(str(number)[::-1])
print(reversed_number)
Enter an integer: 567
765
=== Code Execution Suc

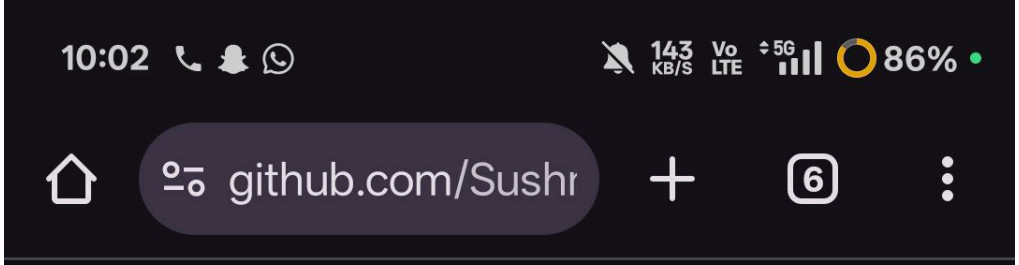
main.py

Run

Output

```
def is_palindrome(input_value):
    input_str = str(input_value)
    if input_str == input_str[::-1]:
        return True
    else:
        return False
input_value = input("Enter a number or string: ")

if is_palindrome(input_value):
    print(f"'{input_value}' is a palindrome.")
else:
    print(f"'{input_value}' is not a palindrome.")

```

```
Enter a number or string: 12345
'12345' is not a palindrome.

=== Code Execution Successful ===
```



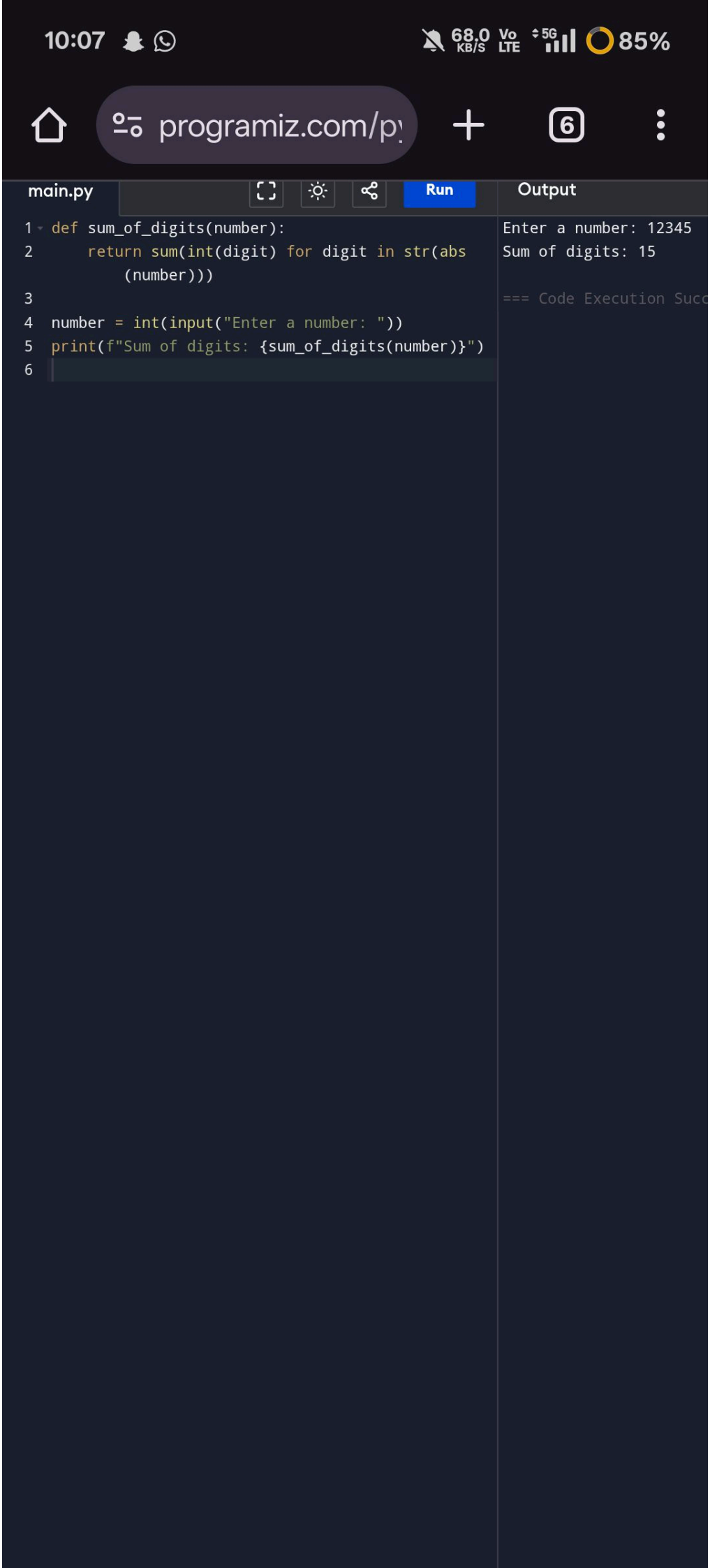
nakg Create evensum.py

12 loc) · 231 Bytes

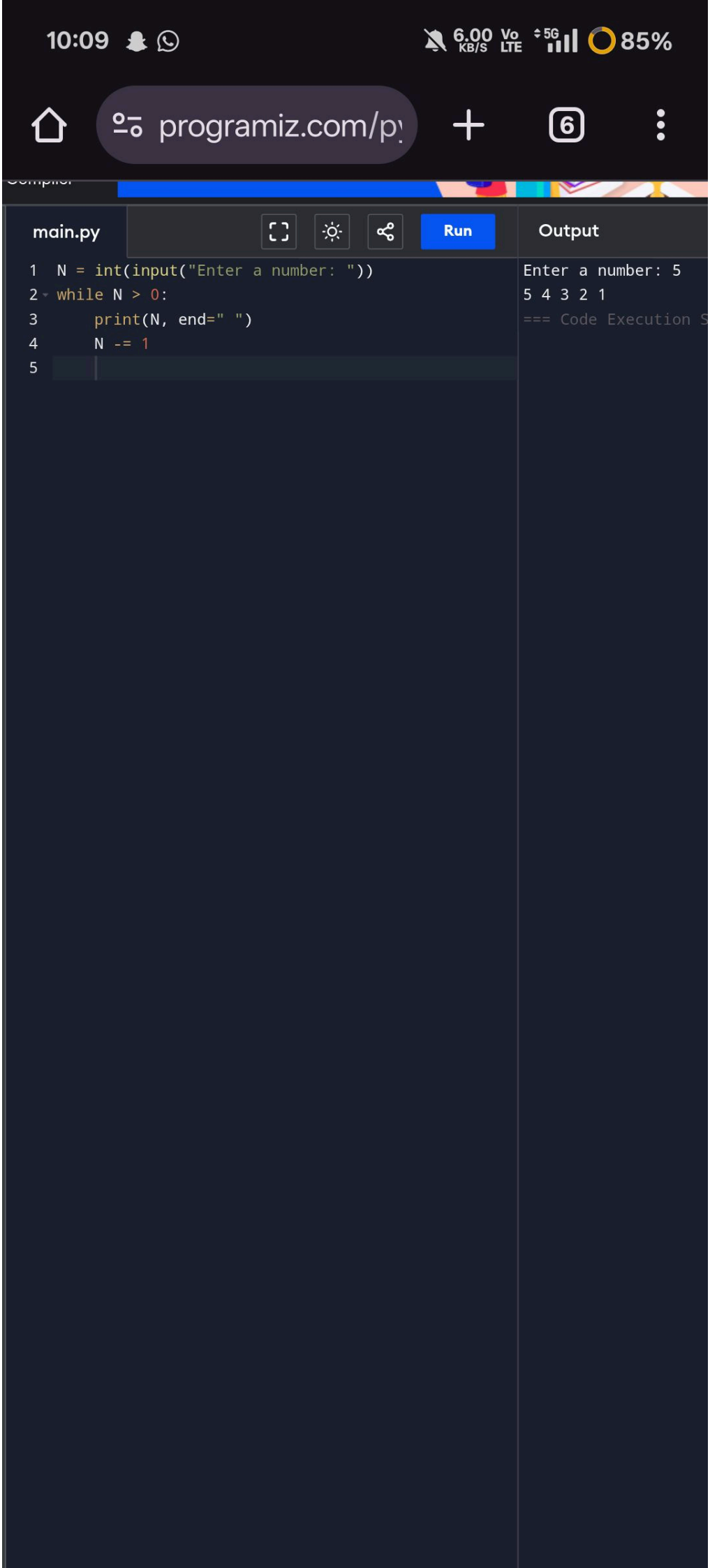
Blame

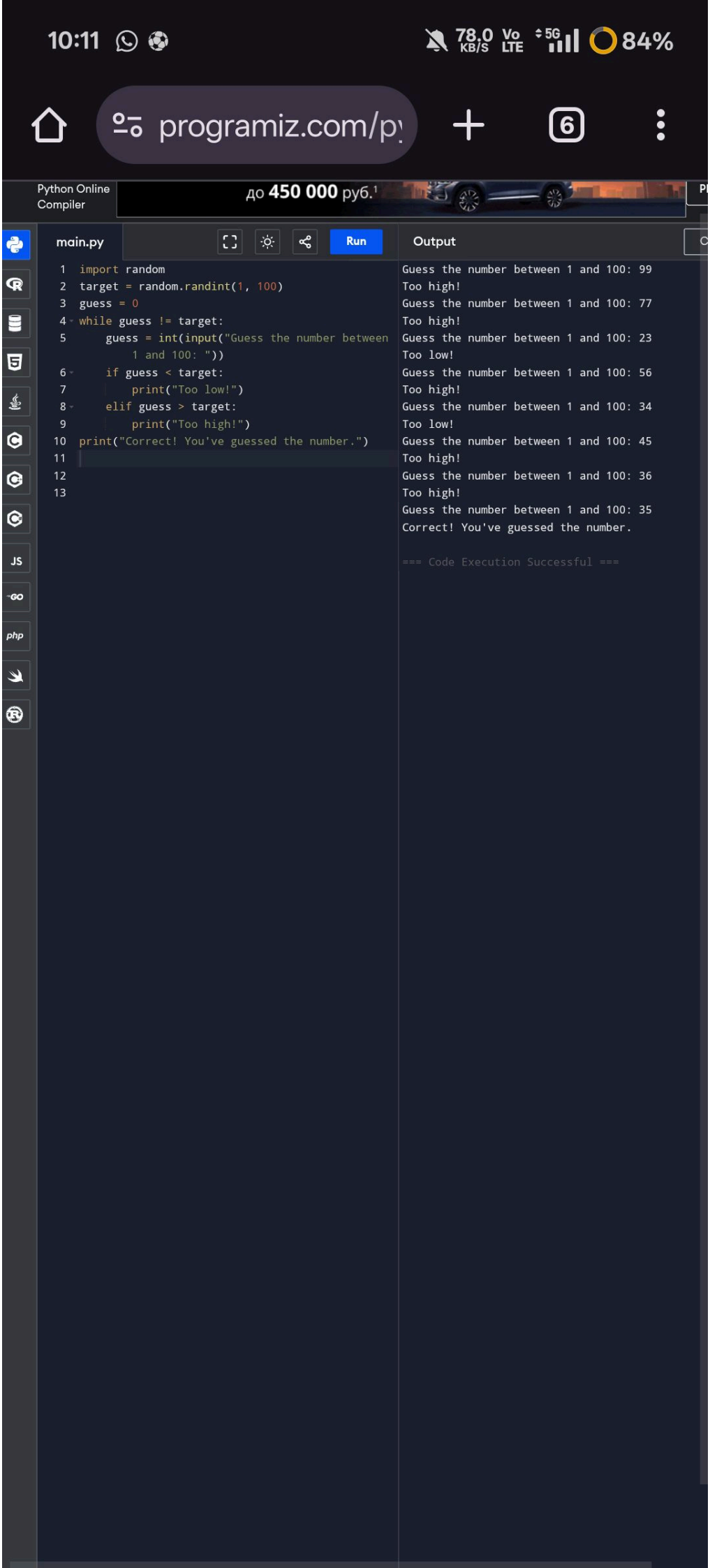
```
#sum of even numbers
a=int(input("enter the starting range"))
b=int(input("entrer the ending range"))
sum=0
for i in range(a,b+1):
  if i%2==0:
    sum=sum+i
print(sum)

output
enter the starting range2
entrer the ending range6
12
```


10:04
11.0 KB/S
Vo LTE
5G
86%
Snapchat • Now
This app is password protected by App Lock.
Hire smarter, hire faster
Learn more
on Online
piler
main.py
Run
Output
def count_digits(number):
return len(str(abs(number)))
number = int(input("Enter an integer: "))
print(f"The number of digits in {number} is
{count_digits(number)}.")
Enter an integer: 23
The number of digits in 23 is 2.
=== Code Execution Successful ===


10:07
68.0 KB/S
Vo LTE
5G
85%
programiz.com/p
6
main.py
Run
Output
1 def sum_of_digits(number):
2 return sum(int(digit) for digit in str(abs
(number)))
3
4 number = int(input("Enter a number: "))
5 print(f"Sum of digits: {sum_of_digits(number)}")
6
Enter a number: 12345
Sum of digits: 15
=== Code Execution Succ


10:08
10.0 KB/S
Vo LTE
5G
85%
programiz.com/p
6
mpiler
main.py
Run
Output
1 def is_armstrong(number):
2 num_str = str(number)
3 num_len = len(num_str)
4 return sum(int(digit) ** num_len for digit in num_str) == number
5
6 number = int(input("Enter a number: "))
7 if is_armstrong(number):
8 print(f"{number} is an Armstrong number.")
9 else:
10 print(f"{number} is not an Armstrong number.")
11
Enter a number: 7
7 is an Armstrong number.
=== Code Execution Success


10:09
6.00 KB/S
Vo LTE
5G
85%
programiz.com/p
6
main.py
Run
Output
1 N = int(input("Enter a number: "))
2 while N > 0:
3 print(N, end=" ")
4 N -= 1
5
Enter a number: 5
5 4 3 2 1
=== Code Execution


10:11
78.0 KB/S
Vo LTE
5G
84%
programiz.com/p
Python Online Compiler
до 450 000 руб.
main.py
Run
Output
import random
target = random.randint(1, 100)
guess = 0
while guess != target:
guess = int(input("Guess the number between 1 and 100: "))
if guess < target:
print("Too low!")
elif guess > target:
print("Too high!")
print("Correct! You've guessed the number.")
Guess the number between 1 and 100: 99
Too high!
Guess the number between 1 and 100: 77
Too high!
Guess the number between 1 and 100: 23
Too low!
Guess the number between 1 and 100: 56
Too high!
Guess the number between 1 and 100: 34
Too low!
Guess the number between 1 and 100: 45
Too high!
Guess the number between 1 and 100: 36
Too high!
Guess the number between 1 and 100: 35
Correct! You've guessed the number.
=== Code Execution Successful ===


10:14
87.0 KB/S
Vo LTE
5G
84%
programiz.com/p
npiler
main.py
Run
Output
1 def is_perfect_number(number):
2 return sum(i for i in range(1, number) if
number % i == 0) == number
3
4 number = int(input("Enter a number: "))
5 if is_perfect_number(number):
6 print(f"{number} is a perfect number.")
7 else:
8 print(f"{number} is not a perfect number.")
9
0
Enter a number: 777
777 is not a perfect number.
=== Code Execution Successful

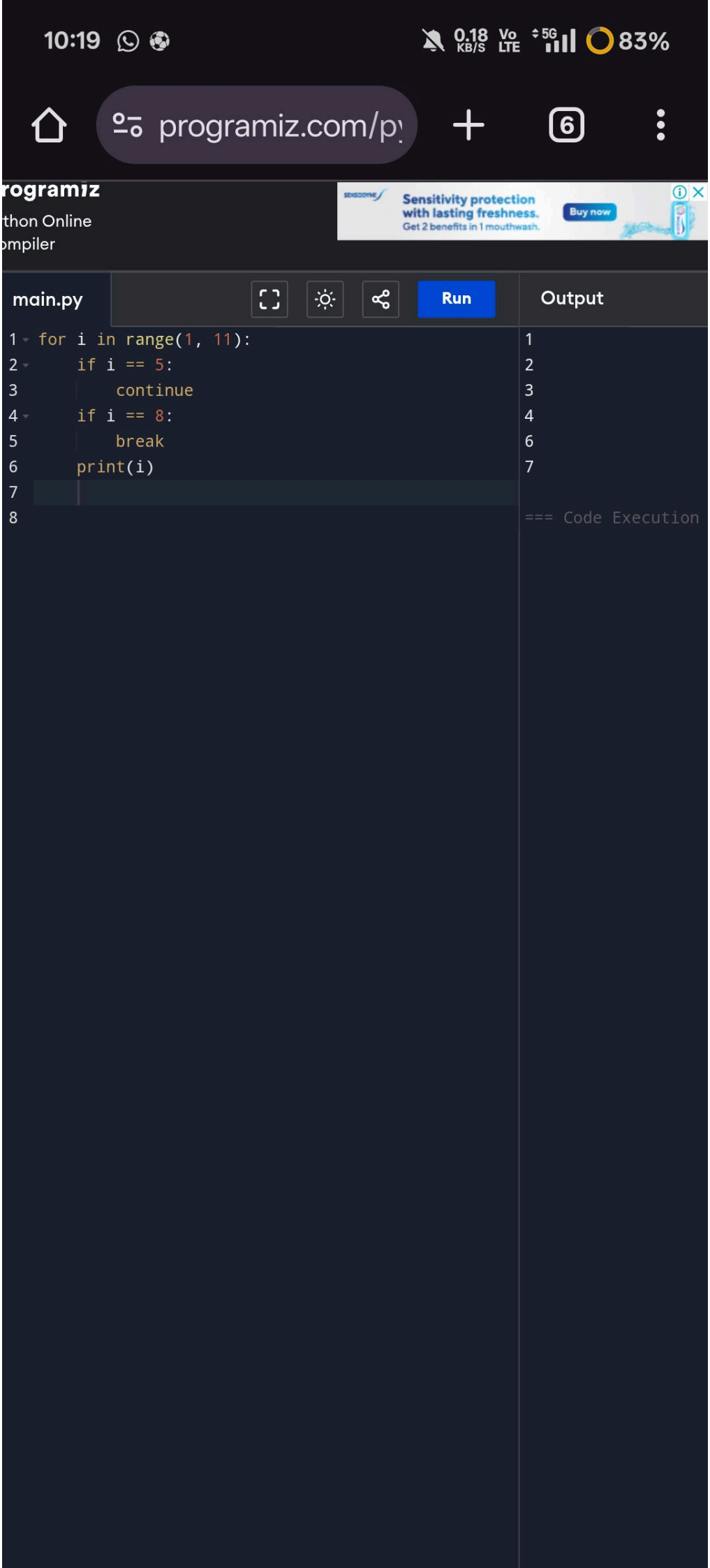
main.py

Run

```python
def print_pyramid(n):
    for i in range(1, n+1):
        print(" " * (n-i) + "*" * (2*i-1))

def print_inverted_pyramid(n):
    for i in range(n, 0, -1):
        print(" " * (n-i) + "*" * (2*i-1))

n = int(input("Enter number of rows for pyramid : "))
print_pyramid(n)
print_inverted_pyramid(n)
```

Output

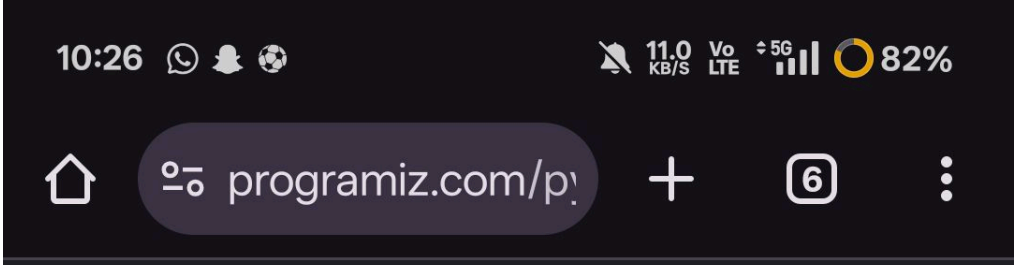
```
Enter number of rows for pyramid: 18
                  *
                 ***
                *****
               *******
              *********
             ***********
            *************
           ***************
          *****************
         *******************
        *********************
       ***********************
      *************************
     ***************************
    *****************************
   *******************************
  *********************************
 ***********************************
*************************************
*************************************
 ***********************************
  *********************************
   *******************************
    *****************************
     ***************************
      *************************
       ***********************
        *********************
         *******************
          *****************
           ***************
            *************
             ***********
              *********
               *******
                *****
                 ***
                  *

=== Code Execution Successful ===
```


10:19
programiz.com/p
rogramiz
thon Online
ompiler
Sensitivity protection
with lasting freshness.
Get 2 benefits in 1 mouthwash.
Buy now
main.py
Run
Output
for i in range(1, 11):
if i == 5:
continue
if i == 8:
break
print(i)
1
2
3
4
6
7
=== Code Execution

main.py Run Output

```python
def gcd(a, b):
    while b:
        a, b = b, a % b
    return a

a = int(input("Enter first number: "))
b = int(input("Enter second number: "))
print(f"GCD: {gcd(a, b)}")

```

```
Enter first number: 6
Enter second number: 8
GCD: 2

=== Code Execution Succe
```



rogramiz
ython Online
ompiler

main.py

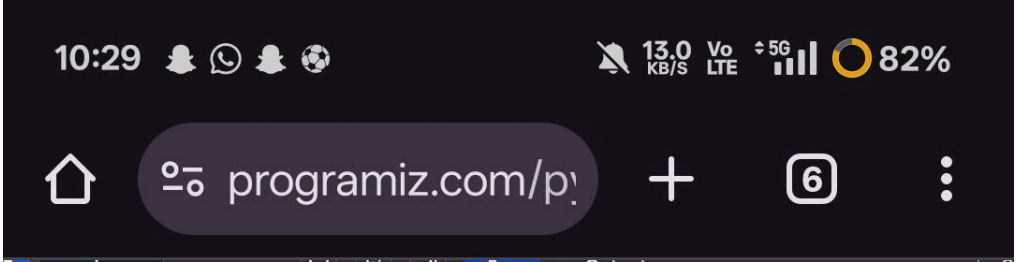
Run

Output

```
def gcd(a, b):
    while b:
        a, b = b, a % b
    return a

def lcm(a, b):
    return abs(a * b) // gcd(a, b)

a = int(input("Enter first number: "))
b = int(input("Enter second number: "))
print(f"LCM: {lcm(a, b)}")
```

```
Enter first number: 4
Enter second number: 8
LCM: 8

=== Code Execution Succes
```


10:28
0.58 KB/S
Vo LTE
5G
82%
programiz.com/p
ogramiz
hon Online
mpiler
7th Jan Aushadhi Diwas
main.py
Run
Output
def sum_of_squares(n):
return sum(i**2 for i in range(1, n+1))
n = int(input("Enter a number: "))
print(f"Sum of squares: {sum_of_squares(n)}")
Enter a number: 3
Sum of squares: 14
=== Code Execution Suc



main.py

Run

Output

```
1
2  def pascal_triangle(n):
3      for i in range(n):
4          print(" " * (n-i-1), end="")
5          num = 1
6          for j in range(i+1):
7              print(num, end=" ")
8              num = num * (i-j) // (j+1)
9          print()
10
11 def floyd_triangle(n):
12     num = 1
13     for i in range(1, n+1):
14         for j in range(i):
15             print(num, end=" ")
16             num += 1
17         print()
18
19 n = int(input("Enter number of rows for
       patterns: "))
20 print("Pascal's Triangle:")
21 pascal_triangle(n)
22 print("Floyd's Triangle:")
23 floyd_triangle(n)
24
```

```
Enter number of rows for patterns: 6
Pascal's Triangle:
     1
    1 1
   1 2 1
  1 3 3 1
 1 4 6 4 1
1 5 10 10 5 1
Floyd's Triangle:
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
16 17 18 19 20 21

=== Code Execution Successful ===
```