main.py

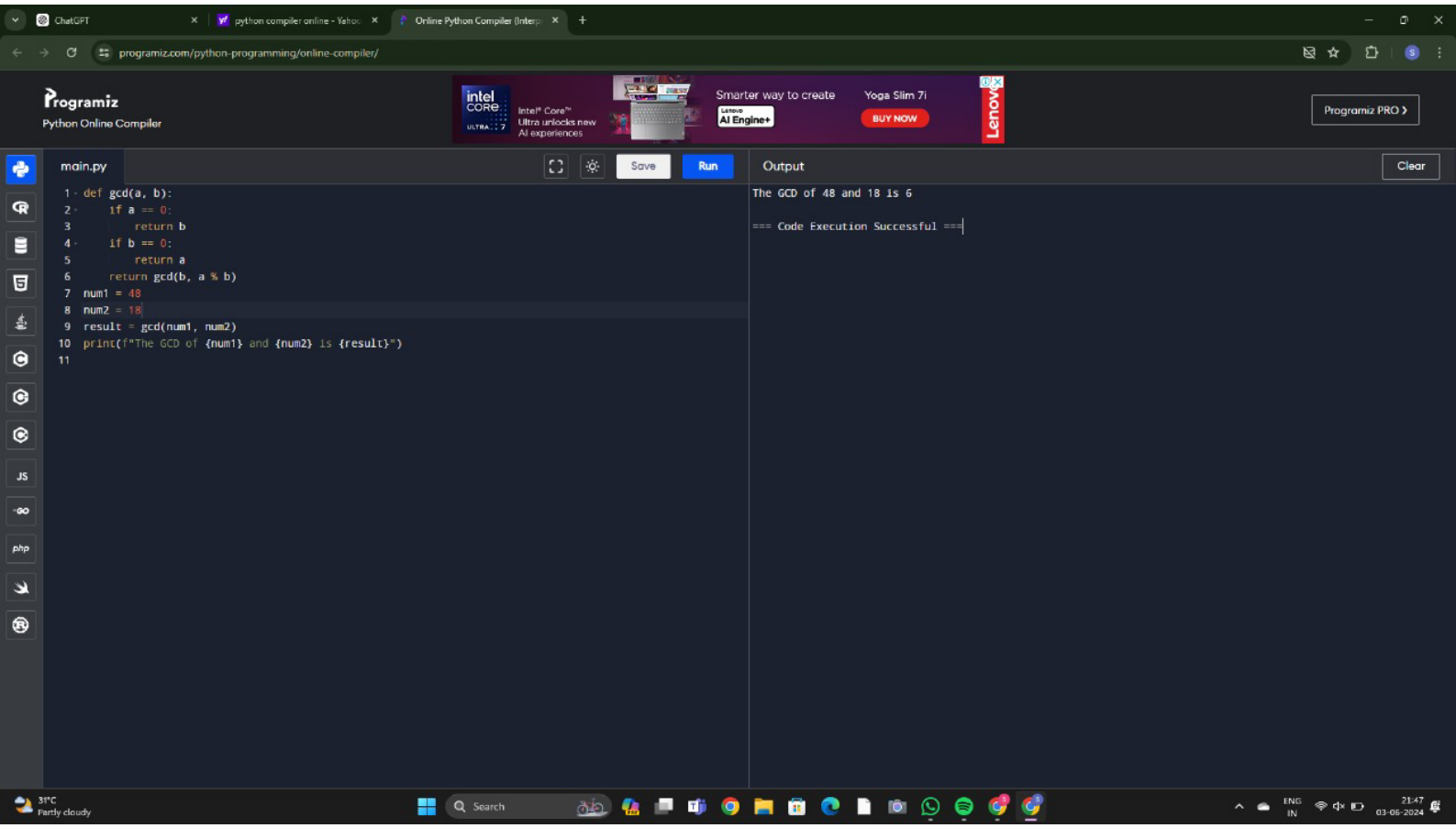
Save Run

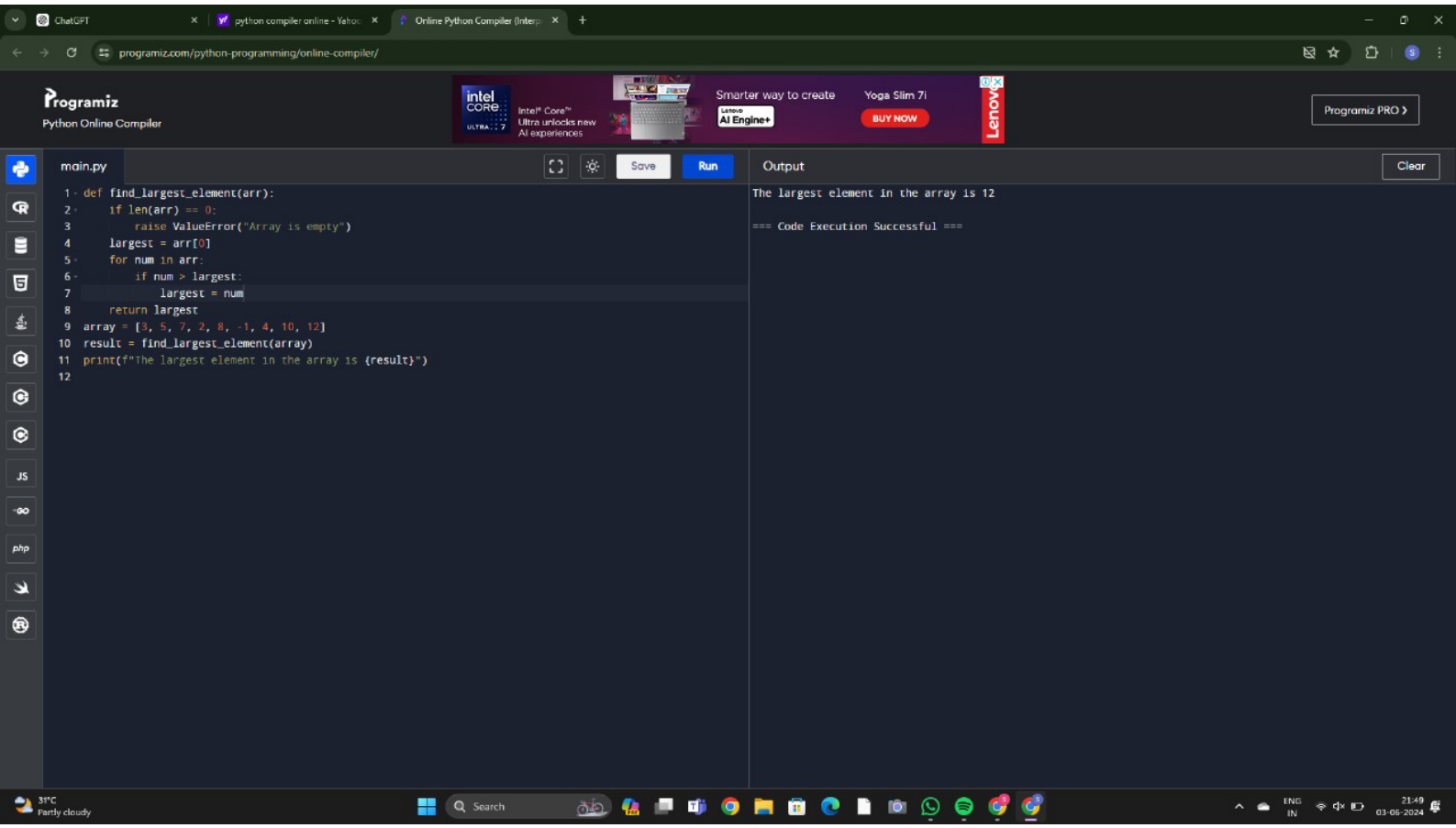
```python
def fibonacci(n):
    if n <= 1:
        return n
    else:
        return fibonacci(n-1) + fibonacci(n-2)

count = int(input("Enter the number of Fibonacci numbers to print : "))

print("Fibonacci Series: ", end="")
for i in range(count):
    print(fibonacci(i), end=" ")
```

Output Clear

```
Enter the number of Fibonacci numbers to print: 5
Fibonacci Series: 0 1 1 2 3
=== Code Execution Successful ===
```


main.py
Save
Run
Output
Clear
def is_armstrong(num, n):
if num == 0:
return 0
return ((num % 10) ** n) + is_armstrong(num // 10, n)
def check_armstrong(num):
n = order(num)
result = is_armstrong(num, n)
return result == num
number = int(input("Enter a number: "))
if check_armstrong(number):
print(number, "is an Armstrong number")
else:
print(number, "is not an Armstrong number")
Enter a number: 123
123 is not an Armstrong number
=== Code Execution Successful ===


ChatGPT
python compiler online - Yahoo
Online Python Compiler (Interp
programiz.com/python-programming/online-compiler/
Programiz
Python Online Compiler
intel CORE ULTRA 7
Intel® Core™ Ultra unlocks new AI experiences
Smarter way to create
Lenovo AI Engine+
Yoga Slim 7i
BUY NOW
Lenovo
Programiz PRO
main.py
Save
Run
Output
Clear
def gcd(a, b):
if a == 0:
return b
if b == 0:
return a
return gcd(b, a % b)
num1 = 48
num2 = 18
result = gcd(num1, num2)
print(f"The GCD of {num1} and {num2} is {result}")
The GCD of 48 and 18 is 6
=== Code Execution Successful ===
31°C
Partly cloudy
Search
ENG
IN
21:47
03-06-2024


ChatGPT
python compiler online - Yaho
Online Python Compiler (Interp
programiz.com/python-programming/online-compiler/
Programiz
Python Online Compiler
intel CORE ULTRA 7
Intel® Core™ Ultra unlocks new AI experiences
Smarter way to create
Lenovo AI Engine+
Yoga Slim 7i
BUY NOW
Lenovo
Programiz PRO
main.py
Save
Run
Output
Clear
def find_largest_element(arr):
if len(arr) == 0:
raise ValueError("Array is empty")
largest = arr[0]
for num in arr:
if num > largest:
largest = num
return largest
array = [3, 5, 7, 2, 8, -1, 4, 10, 12]
result = find_largest_element(array)
print(f"The largest element in the array is {result}")
The largest element in the array is 12
=== Code Execution Successful ===
31°C
Partly cloudy
Search
ENG
IN
21:49
03-06-2024

main.py

Save Run

Output

```python
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n-1)

# Taking input from user
num = int(input("Enter a number: "))

# Checking if the number is negative
if num < 0:
    print("Factorial cannot be found for negative numbers.")
elif num == 0:
    print("The factorial of 0 is 1.")
else:
    print("The factorial of", num, "is", factorial(num))

```

```
Enter a number: 5
The factorial of 5 is 120

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

ChatGPT | python compiler online - Yahoo | Online Python Compiler (Interp

programiz.com/python-programming/online-compiler/

Programiz
Python Online Compiler

Programiz PRO ›

main.py

Save | Run

```
def copy_string_recursive(source):
    if source == "":
        return ""
    return source[0] + copy_string_recursive(source[1:])
original_string = "Hello, World!"
copied_string = copy_string_recursive(original_string)
print(f"Original String: {original_string}")
print(f"Copied String: {copied_string}")

```

Output | Clear

```
Original String: Hello, World!
Copied String: Hello, World!

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

CAD/INR -0.45%

Search

ENG IN

21:53 03-06-2024

main.py

Save Run

```python
def reverse_string(s):
    if len(s) == 0:
        return s
    else:
        return reverse_string(s[1:]) + s[0]

string = input("Enter a string: ")


print("The reverse of the string is:", reverse_string(string))

```

Output Clear

```
Enter a string: amma
The reverse of the string is: amma

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

main.py

Save Run

```python
def is_prime(num, divisor=2):
    if num < 2:
        return False
    if num == 2:
        return True
    if num % divisor == 0:
        return False
    if divisor * divisor > num:
        return True
    return is_prime(num, divisor + 1)

def generate_primes(n, current=2):
    if current > n:
        return
    if is_prime(current):
        print(current)
    generate_primes(n, current + 1)
```

Output

```
Enter the limit to generate prime numbers: 5
Prime numbers up to 5 are:
2
3
5

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

```python
def is_palindrome(s):
    s = s.lower()
    if len(s) <= 1:
        return True
    elif s[0] != s[-1]:
        return False
    else:
        return is_palindrome(s[1:-1])


string = input("Enter a string: ")


if is_palindrome(string):
    print("The string is a palindrome.")
else:
    print("The string is not a palindrome.")
```

```
Enter a string: kavi
The string is not a palindrome.

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

main.py

Save Run

Output

```python
def is_prime(n, i=2):
    if n <= 2:
        return n == 2
    if n % i == 0:
        return False
    if i * i > n:
        return True
    return is_prime(n, i + 1)


num = int(input("Enter a number: "))

if is_prime(num):
    print(num, "is a prime number.")
else:
    print(num, "is not a prime number.")

```

```
Enter a number: 5
5 is a prime number.

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