MANAV RACHNA UNIVERSITY

AIML SUPERVISED LEARNING 3rd SEM

ASSIGNMENT 1

NAME: Abhijeet Kumar Soni

23K23CSUNO1240

AIML 3B

1. Fibonacci Sequence

Write a function to generate the first n Fibonacci numbers using a loop. For example, if n = 6, the output should be [0, 1, 1, 2, 3, 5].

main.py

```python
def fibonacci(n):
    fib_sequence = [0, 1]
    while len(fib_sequence) < n:
        fib_sequence.append(fib_sequence[-1] + fib_sequence[-2])
    return fib_sequence[:n]

n = 6
print(fibonacci(n))
```

Ln: 6, Col: 1

Run Share Command Line Arguments

```
[0, 1, 1, 2, 3, 5]


** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

2. Factorial Calculation

Write a function to compute the factorial of a given number using a loop. For example, factorial(5) should return 120.

main.py

```python
def factorial(n):
    result = 1
    for i in range(2, n + 1):
        result *= i
    return result

n = 5
print(factorial(n))
```

Ln: 9, Col: 1

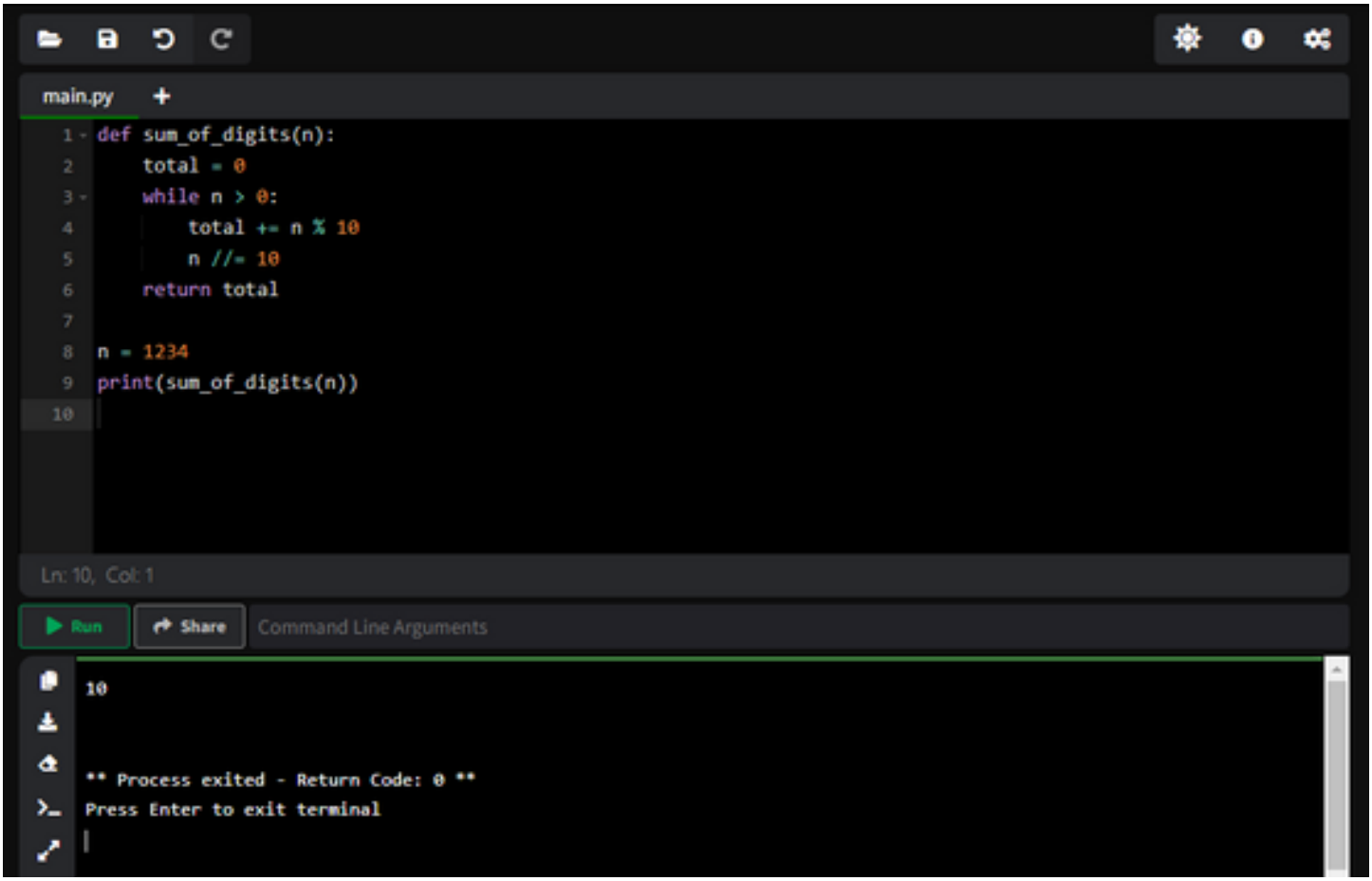
Run Share Command Line Arguments

```
120


** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

3. Sum of Digits

Write a function that takes an integer and returns the sum of its digits. For example, for the number 1234, the output should be 10 (1 + 2 + 3 + 4).



4. Prime Number Check

Write a function to check if a given number is prime using a loop. For example, is_prime(29) should return True, and is_prime(10) should return False.

main.py

```python
def is_prime(n):
    if n <= 1:
        return False
    for i in range(2, int(n**0.5) + 1):
        if n % i == 0:
            return False
    return True

n = 29
print(is_prime(n))
```

Ln: 11, Col: 1

Run | Share | Command Line Arguments

```
True


** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

5. Reverse a String

Write a function to reverse a given string using a loop. For example, for the input "hello", the output should be "olleh".

main.py

```python
def reverse_string(s):
    reversed_str = ''
    for char in s:
        reversed_str = char + reversed_str
    return reversed_str

s = "hello"
print(reverse_string(s))
```

Ln: 9, Col: 1

Run Share Command Line Arguments

```
olleh


** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

6. Palindrome Check

Write a function to check if a given string is a palindrome using a loop. For example, "madam" is a palindrome.

main.py

```python
def is_palindrome(s):
    for i in range(len(s) // 2):
        if s[i] != s[-i-1]:
            return False
    return True

s = "madam"
print(is_palindrome(s))
```

Ln: 9, Col: 1

Run Share Command Line Arguments

```
True


** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

7. Print Multiplication Table

Write a function that prints the multiplication table for a given number n up to 10.

main.py

```python
def multiplication_table(n):
    for i in range(1, 11):
        print(f"{n} x {i} = {n * i}")

n = 5
multiplication_table(n)
```

Ln: 7, Col: 1

Run Share Command Line Arguments

```
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50


** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

8. Find the Largest Number in a List

Write a function that finds the largest number in a list using a loop. For example, given [4, 7, 1, 8, 3], the output should be 8.

main.py

```python
def find_largest(lst):
    largest = lst[0]
    for num in lst[1:]:
        if num > largest:
            largest = num
    return largest

lst = [4, 7, 1, 8, 3]
print(find_largest(lst))
```

Ln: 10, Col: 1

Run Share Command Line Arguments

```
8


** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

10. Print a Pattern

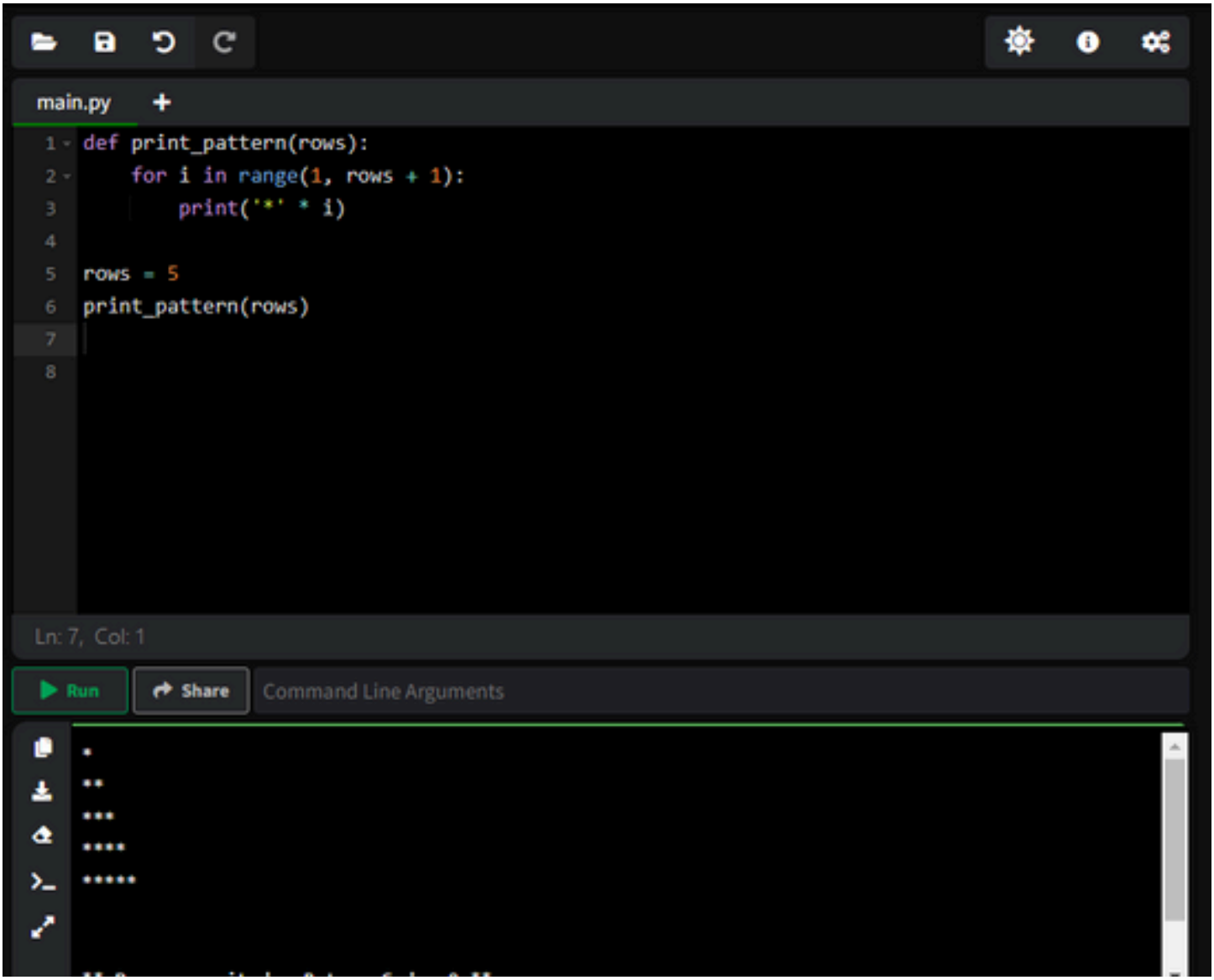
Write a function that prints a pattern of stars based on the number of rows provided. For example, if rows = 5, the output should be:

*

**

***

****



11. Even and Odd Numbers

Write a function that takes a list of integers and returns two lists: one containing all the even numbers and the other containing all the odd numbers.

main.py

```python
def separate_even_odd(numbers):
    even = []
    odd = []
    for num in numbers:
        if num % 2 == 0:
            even.append(num)
        else:
            odd.append(num)
    return even, odd

numbers = [1, 2, 3, 4, 5, 6]
print(separate_even_odd(numbers))
```

Ln: 13, Col: 1

Run Share Command Line Arguments

```
([2, 4, 6], [1, 3, 5])


** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

12. Count Vowels and Consonants

Write a function that takes a string and returns the count of vowels and consonants. For example, for the input "hello world", the output should be {'vowels': 3, 'consonants': 7}.

main.py

```python
def count_vowels_consonants(s):
    vowels = 'aeiouAEIOU'
    count = {'vowels': 0, 'consonants': 0}
    for char in s:
        if char.isalpha():
            if char in vowels:
                count['vowels'] += 1
            else:
                count['consonants'] += 1
    return count

s = "hello world"
print(count_vowels_consonants(s))
```

Ln: 14, Col: 1

Run Share Command Line Arguments

```
{'vowels': 3, 'consonants': 7}

** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

13. Number Pattern

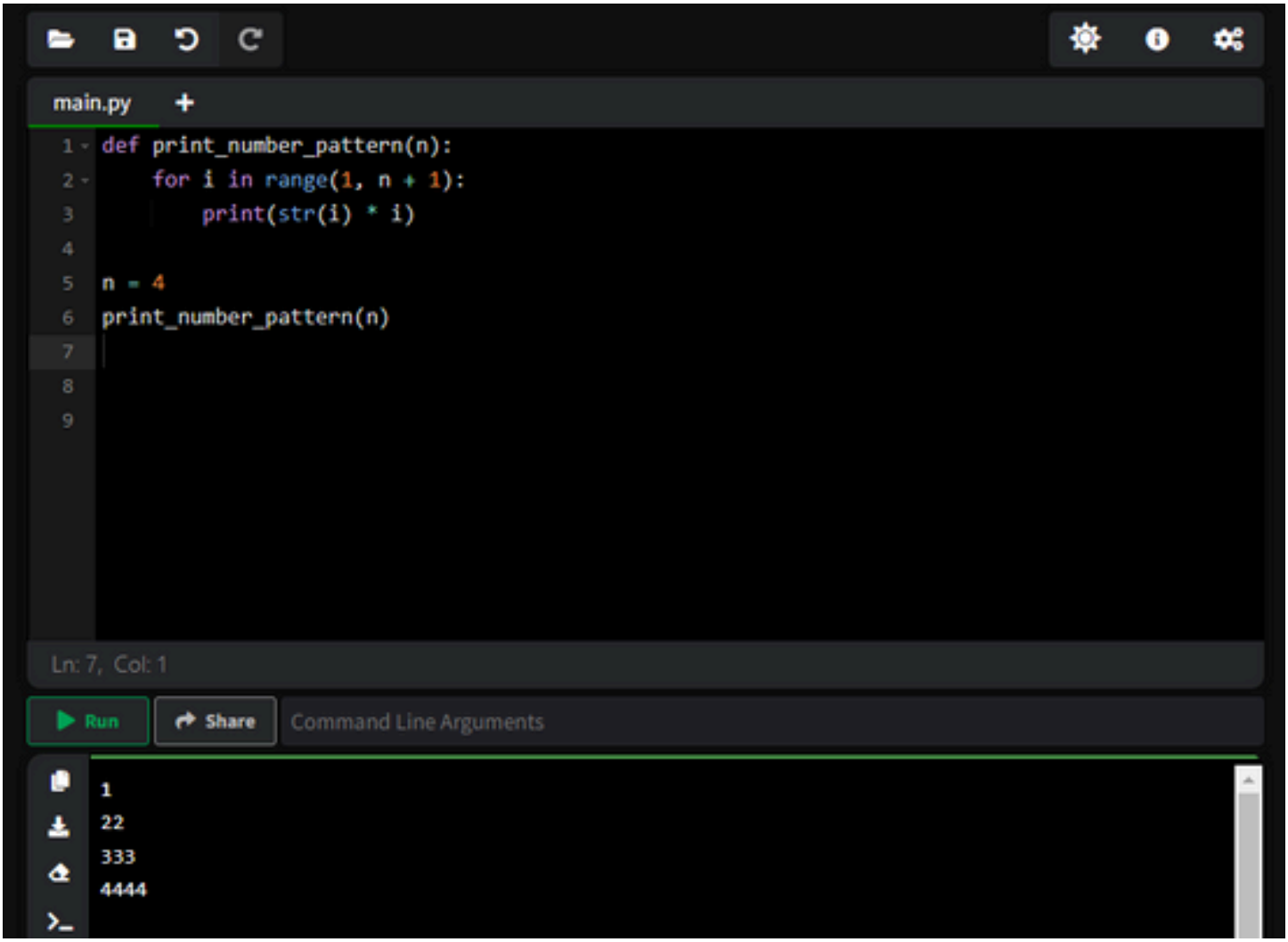
Write a function that prints a number pattern based on a given integer n. For example, for n = 4, the output should be:

1

22

333

4444



14. Multiples of a Number

Write a function that prints all multiples of a given number up to a specified limit. For example, for num = 3 and limit = 20, the output should be 3, 6, 9, 12, 15, 18.

main.py

```python
def print_multiples(num, limit):
    for i in range(num, limit + 1, num):
        print(i, end=", ")

num = 3
limit = 20
print_multiples(num, limit)
```

Ln: 9, Col: 1

Run Share Command Line Arguments

```
3, 6, 9, 12, 15, 18,

** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

15. Sum of Even and Odd Numbers in a List

Write a function that takes a list of integers and returns a tuple with the sum of even numbers and the sum of odd numbers. For example, for the list [1, 2, 3, 4, 5, 6], the output should be (12, 9).

main.py

```python
def sum_even_odd(numbers):
    even_sum = 0
    odd_sum = 0
    for num in numbers:
        if num % 2 == 0:
            even_sum += num
        else:
            odd_sum += num
    return even_sum, odd_sum

numbers = [1, 2, 3, 4, 5, 6]
print(sum_even_odd(numbers))
```

Ln: 13, Col: 1

Run Share Command Line Arguments

```
(12, 9)

** Process exited - Return Code: 0 **
Press Enter to exit terminal
```