**LAB ASSIGNMENT-2.4**

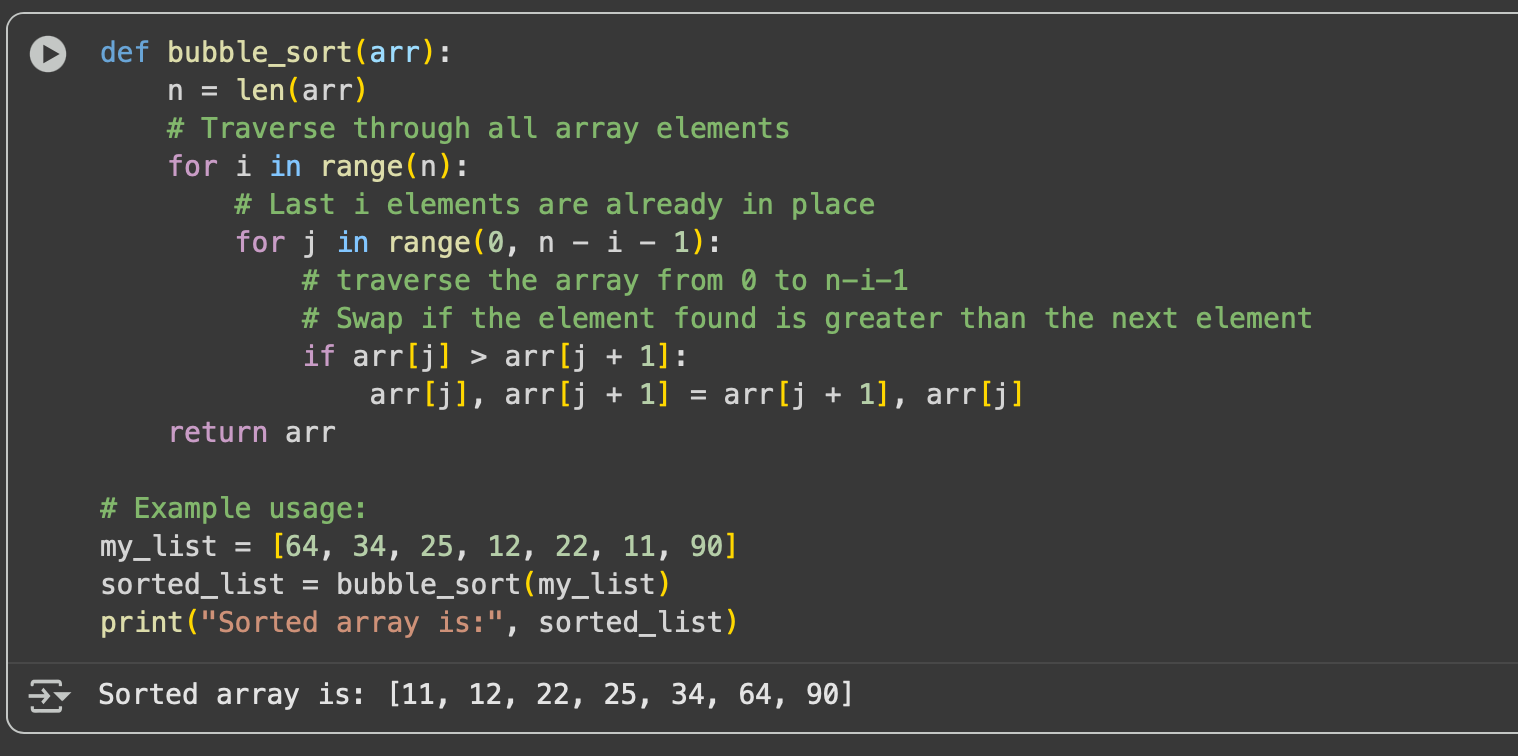
Task Description#1:

Open Google Colab and use Gemini to generate python code that performs sorting of a list using both bubble sort and Python’s built in sort function Compare the two implementations.

Prompt 1:

Create a Python Function which implements Bubble sort algorithm in a list.

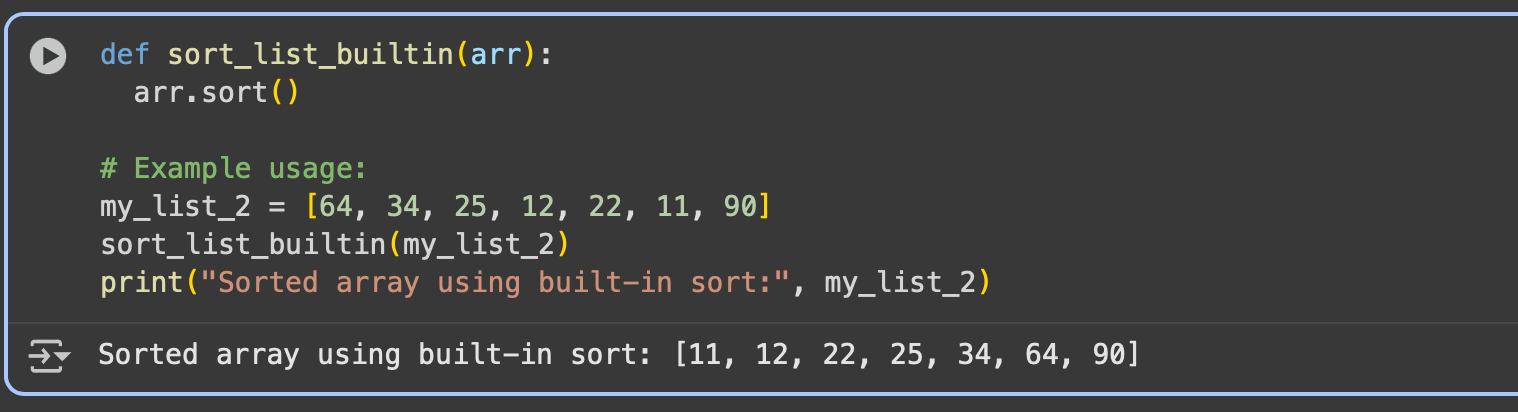
Code & Output:



Prompt 2:

Create a python function for sorting a list using built-in- function sort()

Code & Output:



Task Description #2:

In Colab use Gemini to generate a python function that takes a string and returns ;The number of vowels,the number of consonants,the number of digits in the string

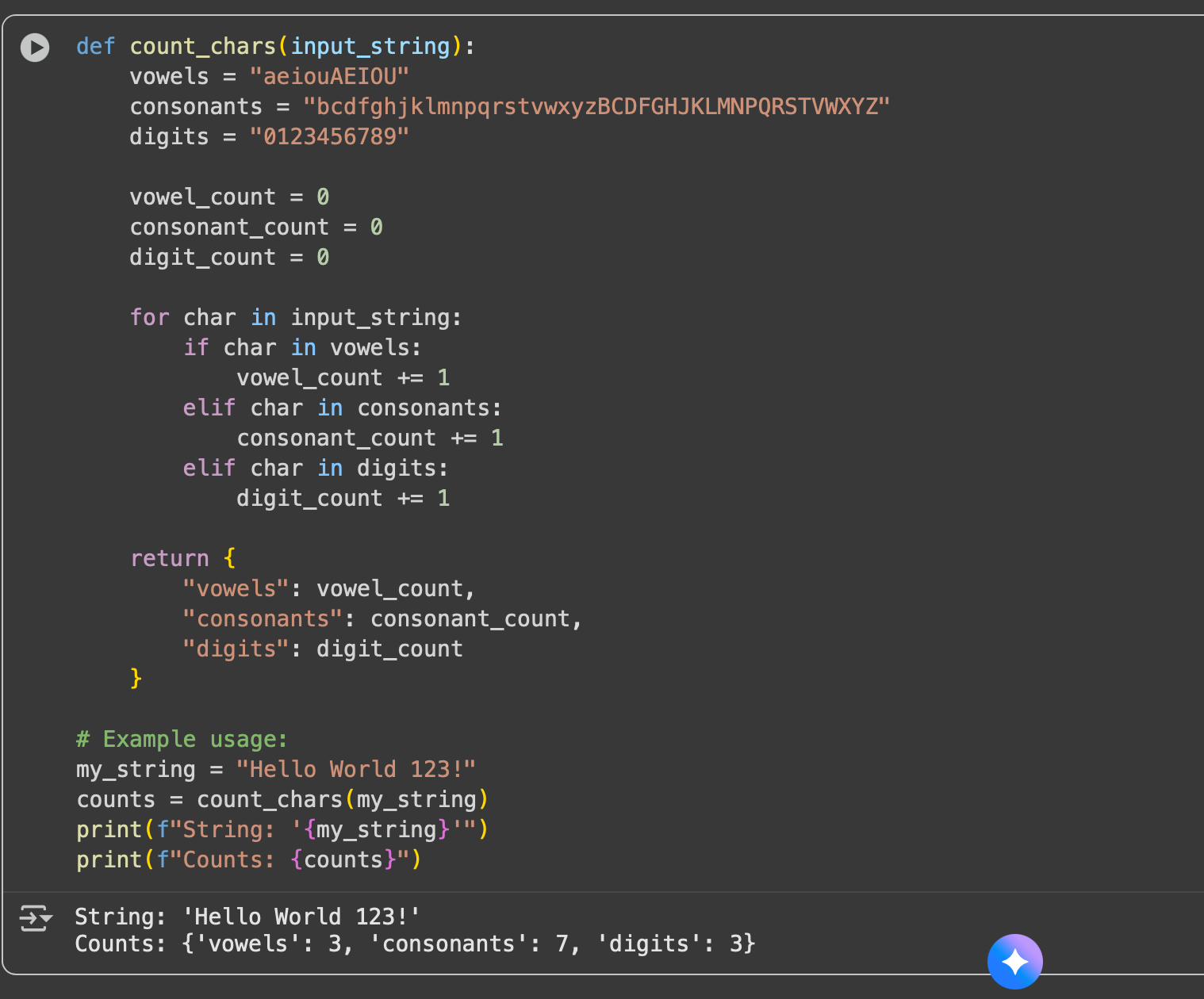
Expected Output # 2:

Complete the function that iterates through characters of a string and counts vowels consonants and digits

Prompt 1:

write a python function that takes a string and returns The number of vowels, consonants ,the number of digits in the string with example.

code & output :



Task Description #3:

Use Gemini to generate a Python program that performs file handling.

Create a text file

Write sample text

Read and display the content

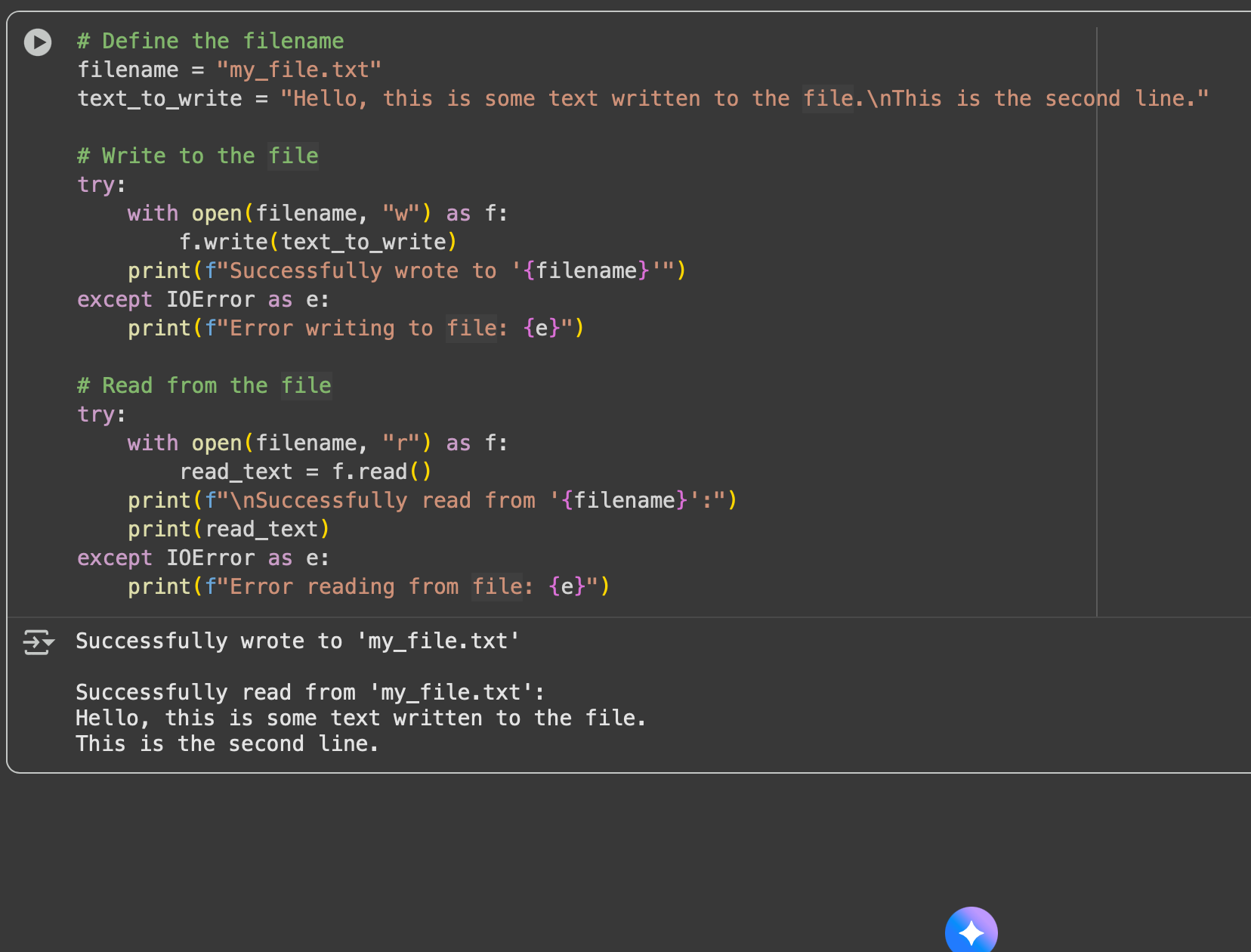
Expected Output:

Functional code that create a .txt file,writes content to it and reads it back

Prompt :

create a python program which performs file handling create a .txt file write text to it and read it back.

Code:



Task Description #4:

Ask Gemini to generate a Python program that implements a simple calculator using functions(add,subtract,multiply,divide)Then asks Gemini to explain how code works.

Expected Output # 4:

Complete calculator code with user input and operation selection

Line-by-line explanation or markdown-style explanation by Gemini

Screenshot of both the code and explanation in Colab

Prompt:

create a python code for a simple calculator using functions(add,sub,mul,division) with user input of number and selction of operations.

Code:

def add(x, y):"""Adds two numbers."""

return x + y

def subtract(x, y):

"""Subtracts two numbers."""

return x - y

def multiply(x, y):

"""Multiplies two numbers."""

return x \* y

def divide(x, y):

"""Divides two numbers. Returns an error message if dividing by zero."""

if y == 0:

return "Error: Division by zero!"

return x / y

print("Select operation:")

print("1. Add")

print("2. Subtract")

print("3. Multiply")

print("4. Divide")

while True:

# Take input from the user

choice = input("Enter choice(1/2/3/4): ")

# Check if choice is one of the four options

if choice in ('1', '2', '3', '4'):

try:

num1 = float(input("Enter first number: "))

num2 = float(input("Enter second number: "))

except ValueError:

print("Invalid input. Please enter a number.")

continue

if choice == '1':

print(num1, "+", num2, "=", add(num1, num2))

elif choice == '2':

print(num1, "-", num2, "=", subtract(num1, num2))

elif choice == '3':

print(num1, "\*", num2, "=", multiply(num1, num2))

elif choice == '4':

result = divide(num1, num2)

print(num1, "/", num2, "=", result)

# Ask the user if they want another calculation

next\_calculation = input("Let's do next calculation? (yes/no): ")

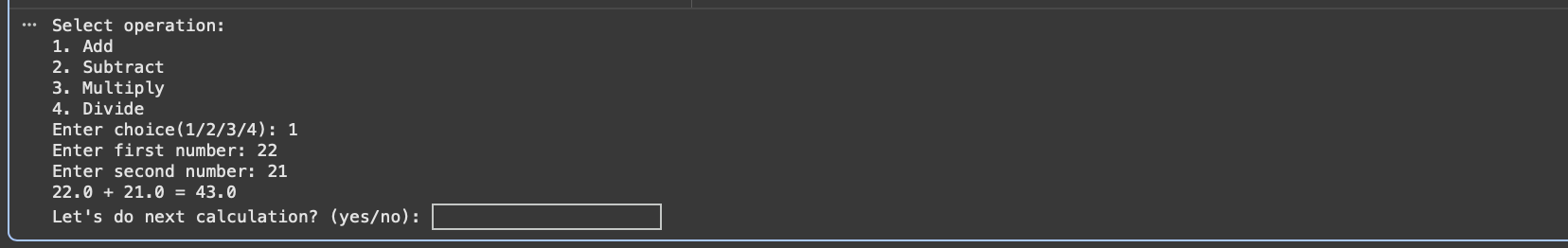
if next\_calculation.lower() != 'yes':

break

else:

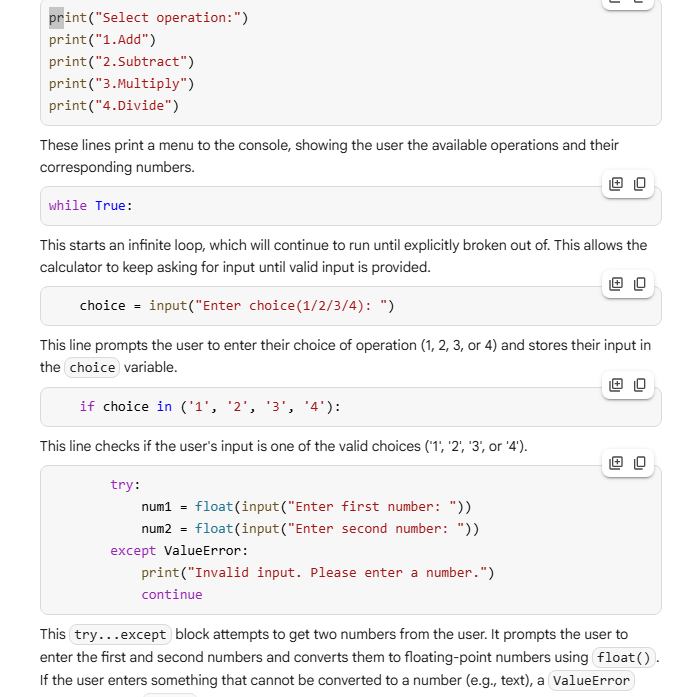
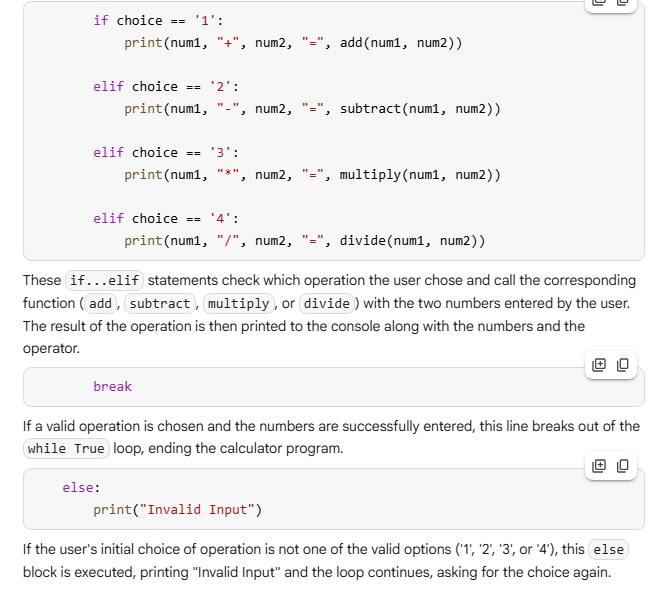
print("Invalid Input")

Output:



Explanation:





Task Description #5:

Use Gemini to create a Python program that checks if a given year is a leap year or not Try different prompts styles and see how Gemini modifies the code suggestions.

Output:

A functional program to check a leap year with a sample input/output

Atleast two versions of code

A short comparision of which version is better and why

Prompt:

create a python program for checking a given year is leap year or not.

Code:

