AIAC LAB ASSIGNMENT 10.2

Name : Navadeep Munugoti

H.No : 2403A52015

Batch : AIAI 02

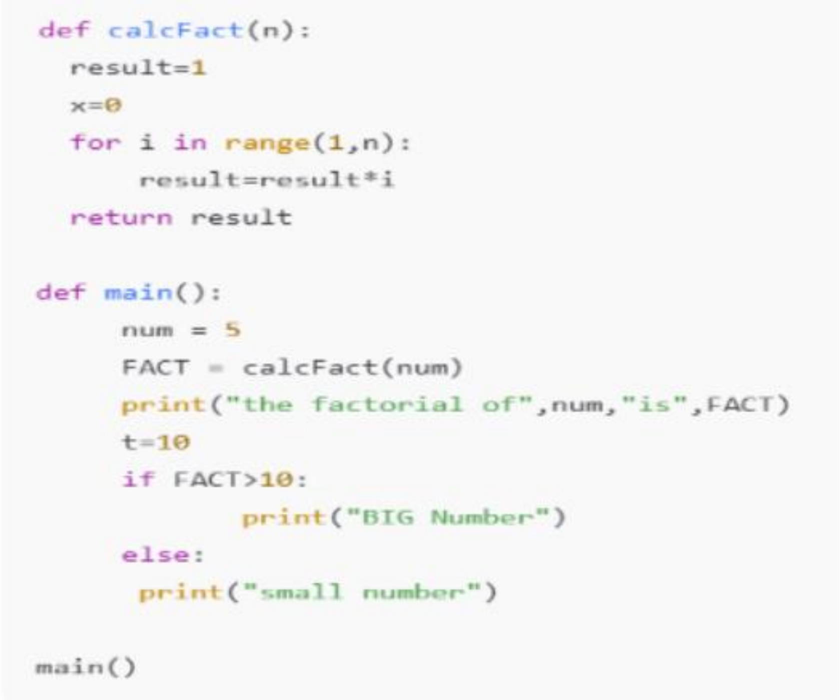
Subject : AI Assisted Coding

Task – 1

Task Description :

Write python program as shown below. Use an AI assistant to review and suggest corrections.

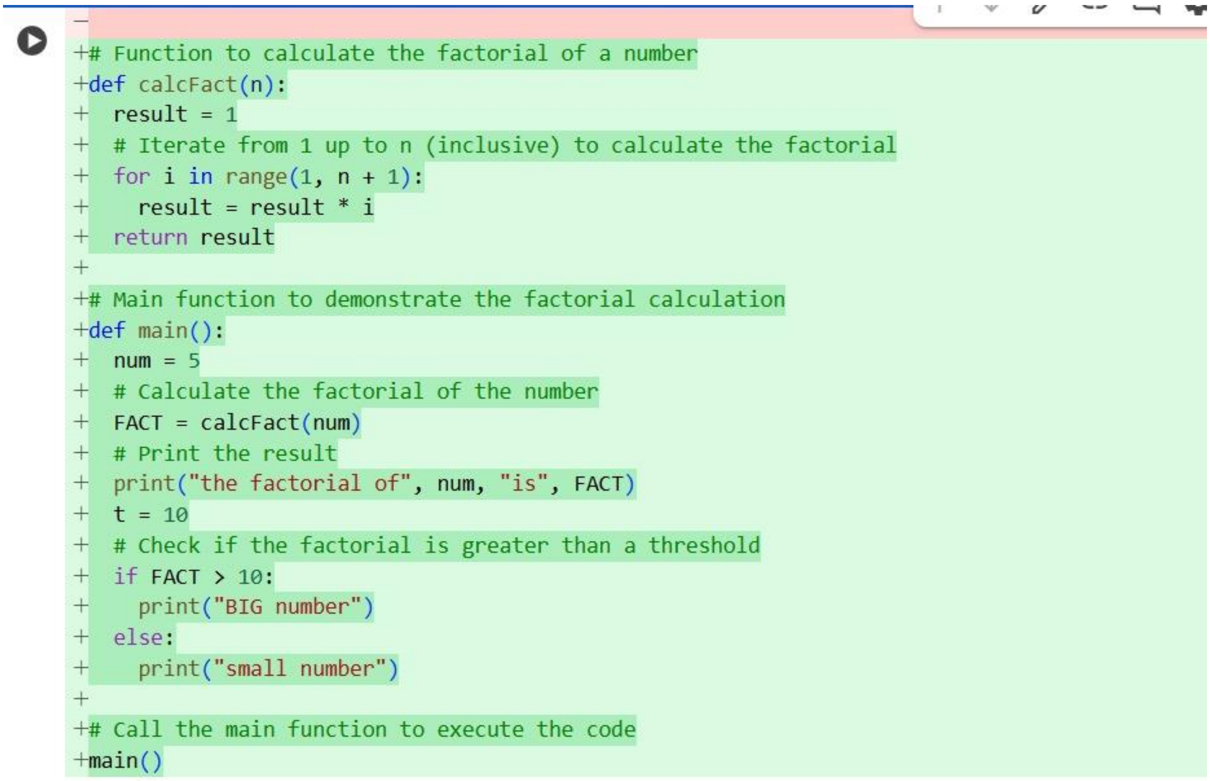
Prompt :



```
def calcFact(n):
  result=1
  x=0
  for i in range(1,n):
       result=result*i
  return result

def main():
      num = 5
      FACT = calcFact(num)
      print("the factorial of",num,"is",FACT)
      t=10
      if FACT>10:
              print("BIG Number")
      else:
       print("small number")

main()
```

Debug the code and add comments for this code

Output:



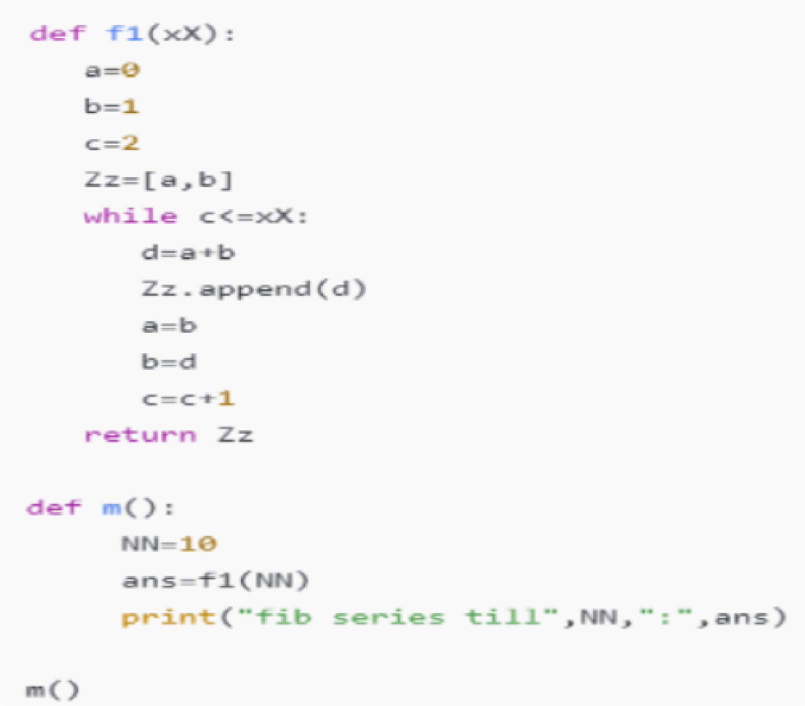
```
-
+# Function to calculate the factorial of a number
+def calcFact(n):
+  result = 1
+  # Iterate from 1 up to n (inclusive) to calculate the factorial
+  for i in range(1, n + 1):
+    result = result * i
+  return result
+
+# Main function to demonstrate the factorial calculation
+def main():
+  num = 5
+  # Calculate the factorial of the number
+  FACT = calcFact(num)
+  # Print the result
+  print("the factorial of", num, "is", FACT)
+  t = 10
+  # Check if the factorial is greater than a threshold
+  if FACT > 10:
+    print("BIG number")
+  else:
+    print("small number")
+
+# Call the main function to execute the code
+main()
```

Task – 2

Task Description :

Write the Python code for Fibonacci as shown below and execute. Ask AI to improve variable names, add comments, and apply PEP8 formatting (cleaned up).

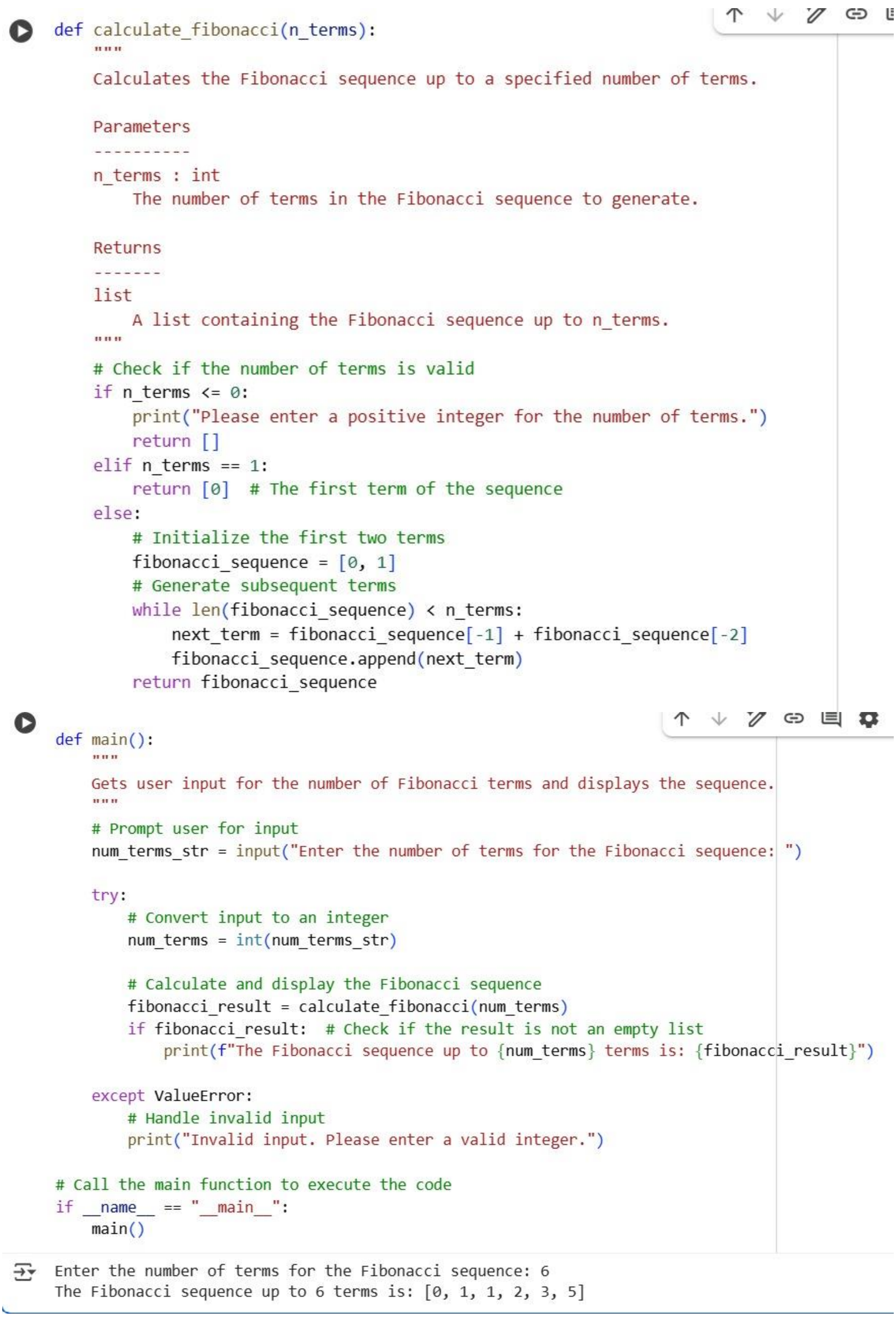
Prompt :



```
def f1(xX):
    a=0
    b=1
    c=2
    Zz=[a,b]
    while c<=xX:
        d=a+b
        Zz.append(d)
        a=b
        b=d
        c=c+1
    return Zz

def m():
      NN=10
      ans=f1(NN)
      print("fib series till",NN,":",ans)

m()
```

Improve variable names, add comments, and apply PEP8 formatting (cleaned up).

Output :



```python
def calculate_fibonacci(n_terms):
    """
    Calculates the Fibonacci sequence up to a specified number of terms.

    Parameters
    ----------
    n_terms : int
        The number of terms in the Fibonacci sequence to generate.

    Returns
    -------
    list
        A list containing the Fibonacci sequence up to n_terms.
    """
    # Check if the number of terms is valid
    if n_terms <= 0:
        print("Please enter a positive integer for the number of terms.")
        return []
    elif n_terms == 1:
        return [0]  # The first term of the sequence
    else:
        # Initialize the first two terms
        fibonacci_sequence = [0, 1]
        # Generate subsequent terms
        while len(fibonacci_sequence) < n_terms:
            next_term = fibonacci_sequence[-1] + fibonacci_sequence[-2]
            fibonacci_sequence.append(next_term)
        return fibonacci_sequence
```

```python
def main():
    """
    Gets user input for the number of Fibonacci terms and displays the sequence.
    """
    # Prompt user for input
    num_terms_str = input("Enter the number of terms for the Fibonacci sequence: ")

    try:
        # Convert input to an integer
        num_terms = int(num_terms_str)

        # Calculate and display the Fibonacci sequence
        fibonacci_result = calculate_fibonacci(num_terms)
        if fibonacci_result:  # Check if the result is not an empty list
            print(f"The Fibonacci sequence up to {num_terms} terms is: {fibonacci_result}")

    except ValueError:
        # Handle invalid input
        print("Invalid input. Please enter a valid integer.")

# Call the main function to execute the code
if __name__ == "__main__":
    main()
```

```
Enter the number of terms for the Fibonacci sequence: 6
The Fibonacci sequence up to 6 terms is: [0, 1, 1, 2, 3, 5]
```

Task – 3

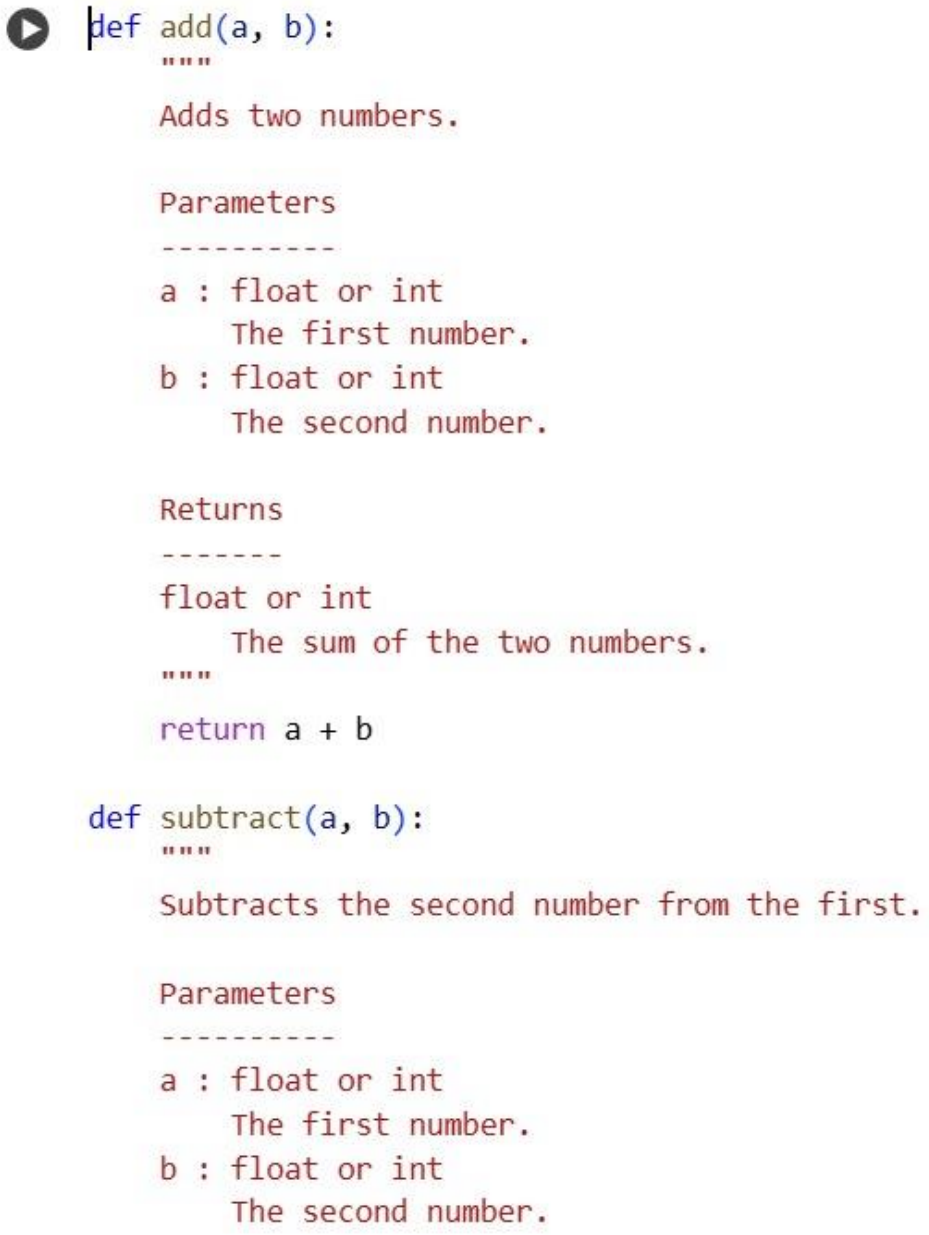
Task Description:

Write a Python script with 3–4 functions (e.g., calculator: add, subtract, multiply, divide). Incorporate manual docstring in code with NumPy Style.

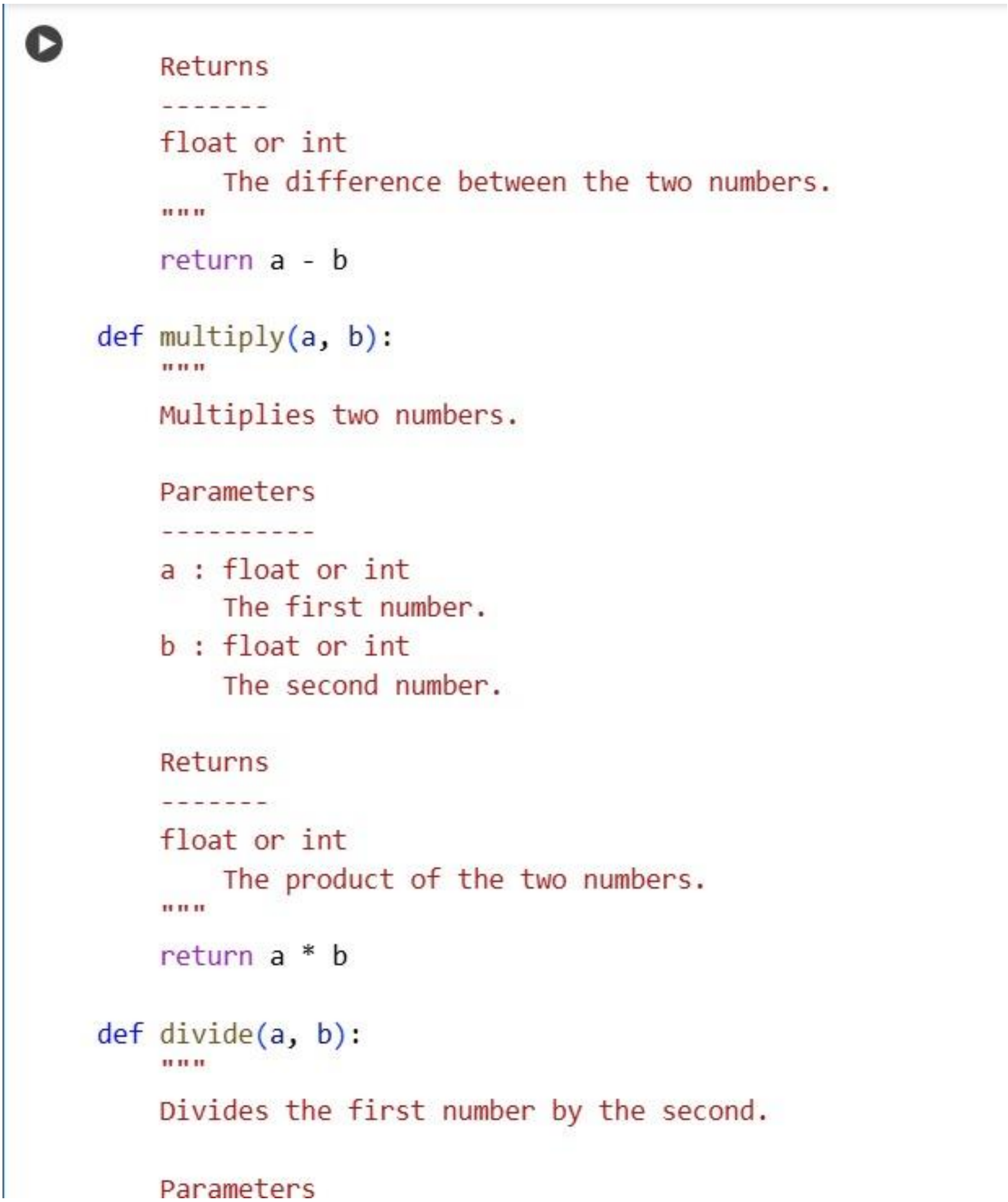
Prompt :

Write a Python script with 3–4 functions (e.g., calculator: add, subtract, multiply, divide). Incorporate manual docstring in code with NumPy Style.

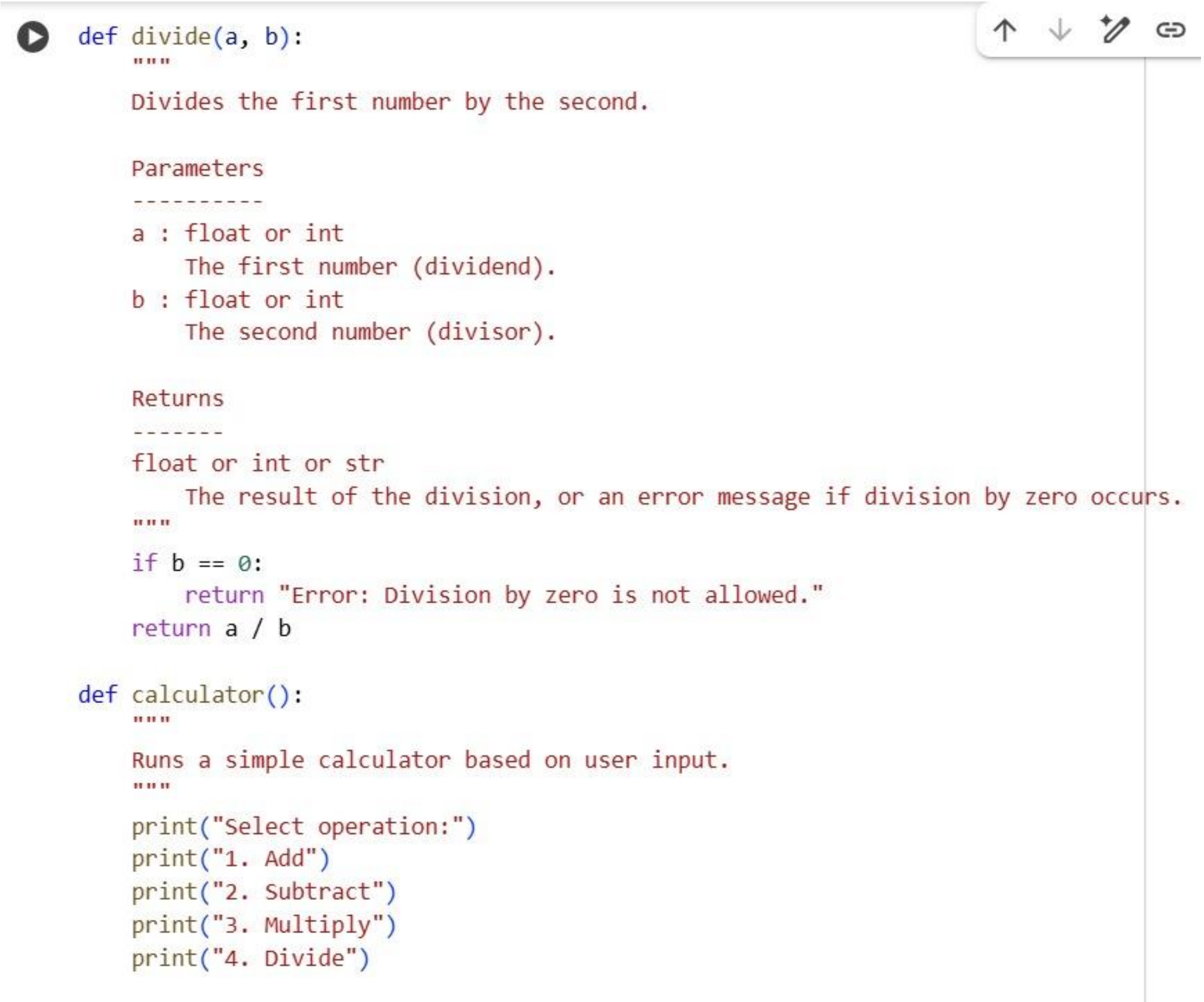
Output:



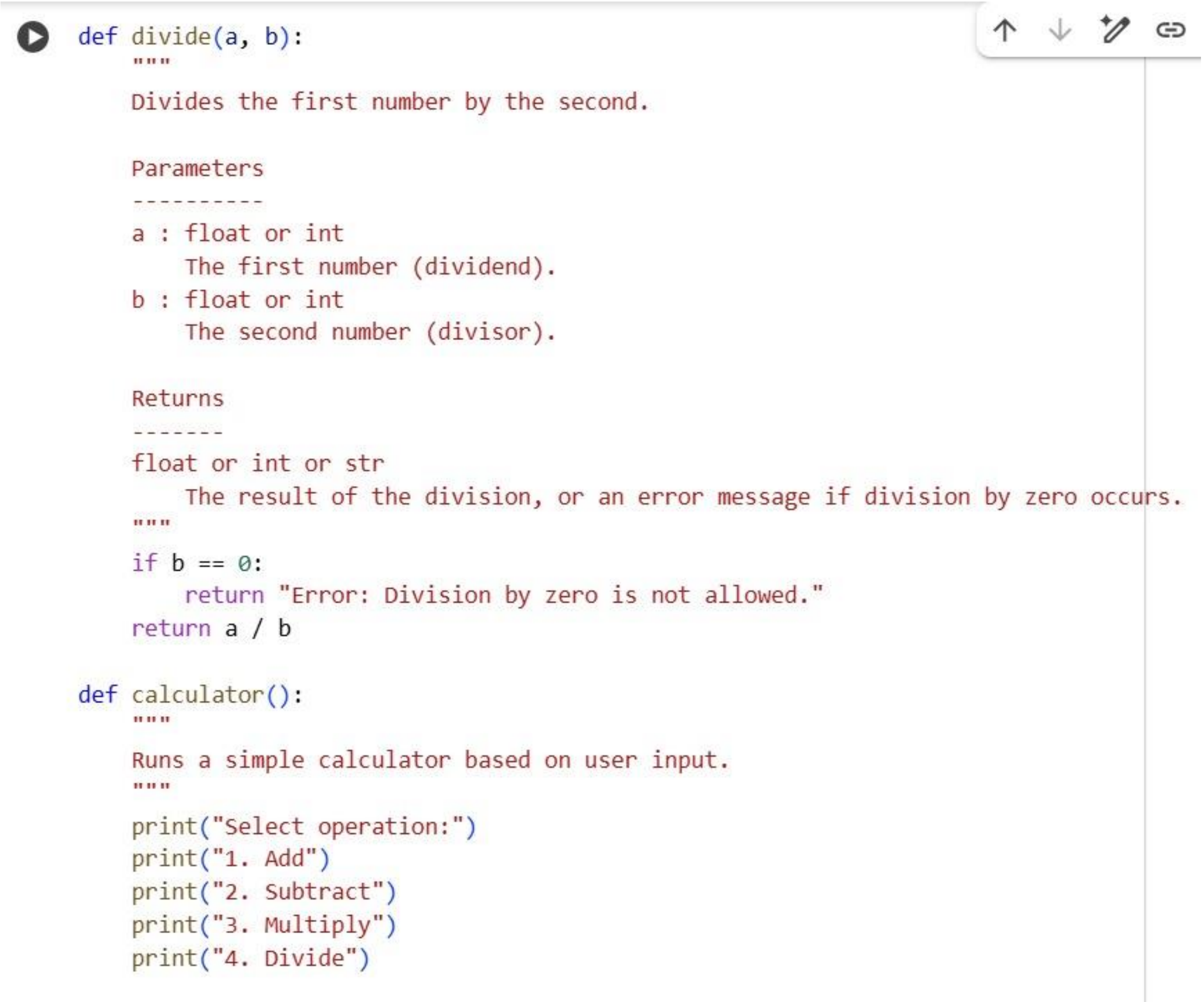
```
def add(a, b):
    """
    Adds two numbers.

    Parameters
    ----------
    a : float or int
        The first number.
    b : float or int
        The second number.

    Returns
    -------
    float or int
        The sum of the two numbers.
    """
    return a + b

def subtract(a, b):
    """
    Subtracts the second number from the first.

    Parameters
    ----------
    a : float or int
        The first number.
    b : float or int
        The second number.
```



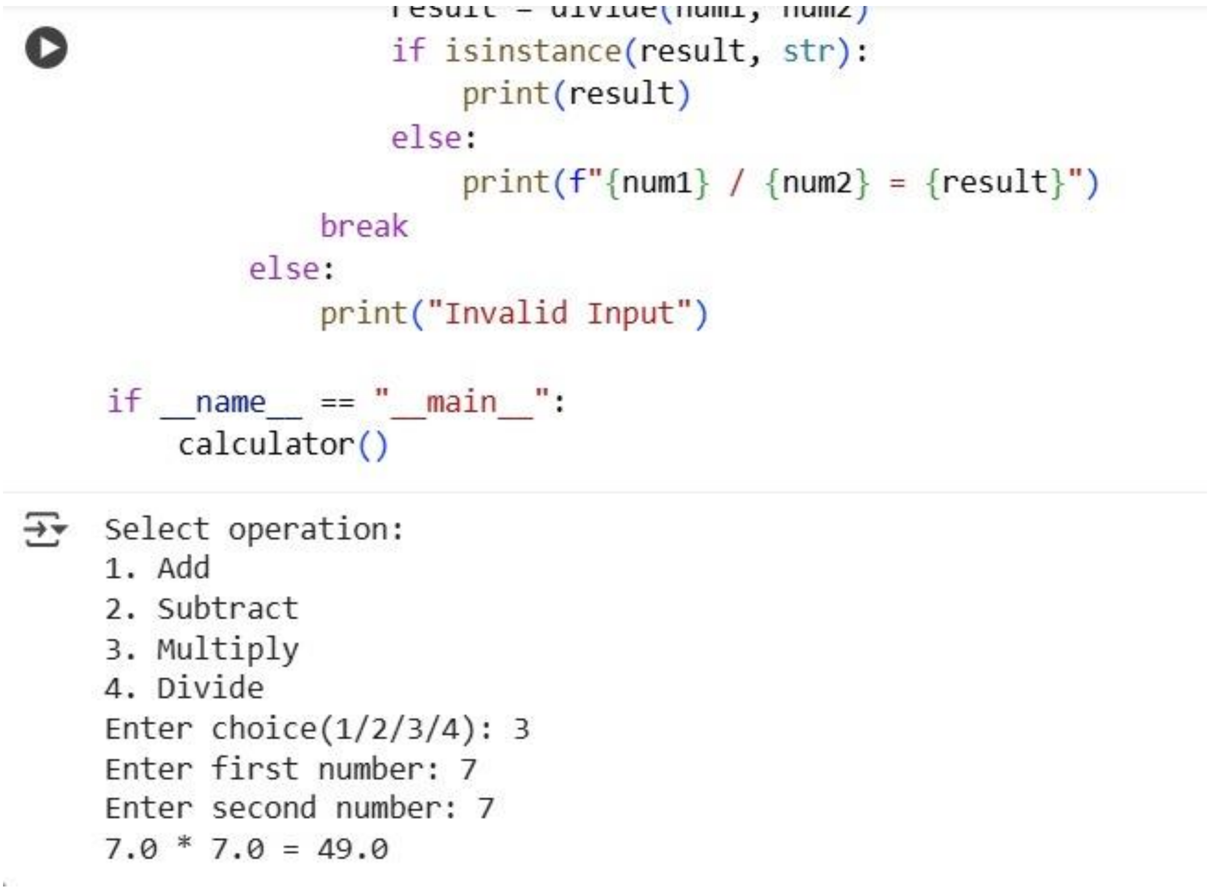
```
    Returns
    -------
    float or int
        The difference between the two numbers.
    """
    return a - b

def multiply(a, b):
    """
    Multiplies two numbers.

    Parameters
    ----------
    a : float or int
        The first number.
    b : float or int
        The second number.

    Returns
    -------
    float or int
        The product of the two numbers.
    """
    return a * b

def divide(a, b):
    """
    Divides the first number by the second.

    Parameters
```



```python
def divide(a, b):
    """
    Divides the first number by the second.

    Parameters
    ----------
    a : float or int
        The first number (dividend).
    b : float or int
        The second number (divisor).

    Returns
    -------
    float or int or str
        The result of the division, or an error message if division by zero occurs.
    """
    if b == 0:
        return "Error: Division by zero is not allowed."
    return a / b

def calculator():
    """
    Runs a simple calculator based on user input.
    """
    print("Select operation:")
    print("1. Add")
    print("2. Subtract")
    print("3. Multiply")
    print("4. Divide")
```



```python
def divide(a, b):
    """
    Divides the first number by the second.

    Parameters
    ----------
    a : float or int
        The first number (dividend).
    b : float or int
        The second number (divisor).

    Returns
    -------
    float or int or str
        The result of the division, or an error message if division by zero occurs.
    """
    if b == 0:
        return "Error: Division by zero is not allowed."
    return a / b

def calculator():
    """
    Runs a simple calculator based on user input.
    """
    print("Select operation:")
    print("1. Add")
    print("2. Subtract")
    print("3. Multiply")
    print("4. Divide")
```



```
                result = divide(num1, num2)
                if isinstance(result, str):
                    print(result)
                else:
                    print(f"{num1} / {num2} = {result}")
            break
        else:
            print("Invalid Input")

if __name__ == "__main__":
    calculator()
```

```
Select operation:
1. Add
2. Subtract
3. Multiply
4. Divide
Enter choice(1/2/3/4): 3
Enter first number: 7
Enter second number: 7
7.0 * 7.0 = 49.0
```