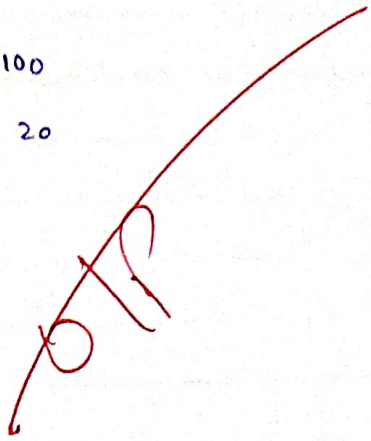
1.	23/07/2025	Running Python Script And Various Expressions in an Interactive Interpreter.	15	6/8/25

Output:-

```
Enter first value: 100
Enter second value: 20
Addition: 120.0
Subtraction: 80.0
Multiplication: 2000.0
Division: 5.0
```



28/7/25

Task-1: Running Python Script And Various Expressions in an Interactive Interpreter

(a) Perform Basic Mathematical Computations.

Aim:-

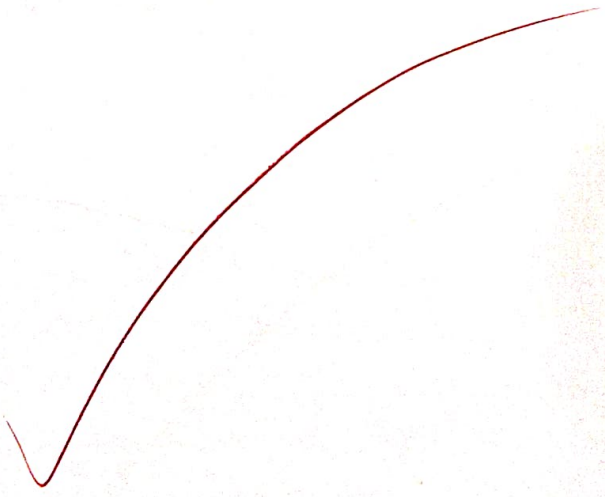
To write a Python program that accepts two numerical inputs and perform addition, subtraction, multiplication, and division operations.

Algorithm:-

1. Start the program.
2. Accept two numerical inputs from the user.
3. perform:
   - Addition
   - Subtraction
   - Multiplication
   - Division (if second number is not zero).
4. Display the results.
5. End the program.

Program:-

```
num1 = float(input("Enter first value:"))
num2 = float(input("Enter second value:"))
print("Addition:", num1 + num2)
print("Subtraction:", num1 - num2)
print("Multiplication:", num1 * num2)
print("Division:", num1 / num2)
```

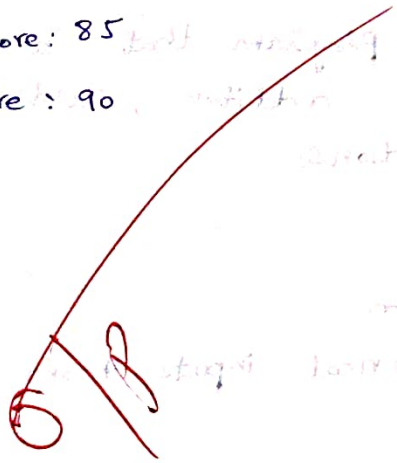


Result:

The program successfully performed all arithmetic operations on the given inputs and displayed the results.

Output:-

```
Enter first score: 85
Enter second score : 90
a > b: False
a < b: True
a == b: False
a != b: True
a >= b: False
a <= b: True
```



3/7/25 (b) Evaluate Relational Expressions.

Aim:-

To develop a python program that compares two numeric values using relational operators and displays the result of each comparison.

Algorithm:-

1. Start the program.
2. Accept two numbers from the user.
3. Apply the following relational operators:
   - Greater than (>)
   - less than (<)
   - Equal to (==)
   - Not equal to (!=)
   - Greater than or equal to (>=)
   - Less than or equal to (<=)
4. Display the results.
5. End the program.

Program:-

```
a = float (input ("Enter first score :"))
b = float (input ("Enter second score :"))
print ("a >b:", a >b)
print ("a <b:", a <b)
print ("a==b:", a == b)
print ("a!=b:", a != b)
print ("a>=b:", a >= b)
print ("a<=b:", a <= b)
```

Result:-

The program correctly evaluated all the relational expressions between the two given inputs.

Output:-

```
Enter marks for Test 1: 45
Enter marks for Test 2: 38
Enter marks for Test 3: 42
Passed all tests: False
Passed at least one test: True
Failed all tests: False
```

08

13/7/25 (c) Check Logical Conditions Across Multiple Inputs.

Aim:-

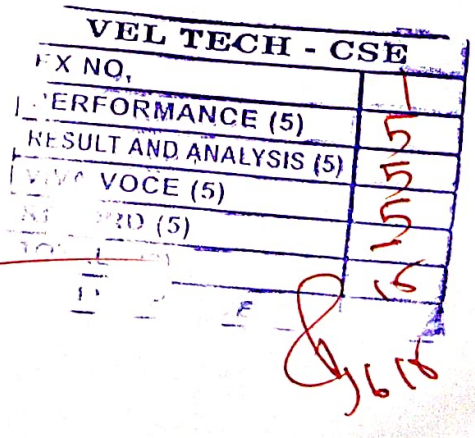
To create a Python program that uses logical operators (and, or, not) to evaluate conditions across three test scores

Algorithm:-

1. Start the program.
2. Accept three test scores from the user.
3. Use logical operators to evaluate:
   - If the candidate passed all tests (and)
   - If the candidate passed at least one test (or)
   - If the candidate failed all tests (not).
4. Display the results.
5. End the program.

Program:-

```
test 1 = int (input ("Enter marks for Test 1:"))
test 2 = int (input ("Enter marks for Test 2:"))
test 3 = int (input ("Enter marks for Test 3:"))
print ("Passed all tests:", test 1 >40 and test 2 >40 and
                                                test 3 >40)
print ("Passed at least one test:", test 1 > 40 or test 2 >40
                                                or test 3 >40)
print ("Failed all tests:", not test 1 >40 or test 2 > 40 or
                                                test 3 >40))
```



Result:-

The program effectively evaluated logical expressions and correctly identified pass/fail conditions based on test scores.