

Date :- 23/7/25

Task 1:- Running Python script and various expressions in an interactive interpreter.

Aim:- To run python script and various expressions in an interactive interpreter that accepts two numerical inputs and performs addition, subtraction, multiplication and division operation.

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)
```

output:

enter first value : 100
enter second value : 20
Addition : 120.0

Subtraction : 80.0

Multiplication : 2000.0

Division : 5.0

Modulo operation : 0

round off

margin left from

gets more closer towards zero if

more

multiple

multiple

points of view

fractional part

case for the integer divide to be carried

either with negative

```
Print ("Multiplication:", num1 * num2)
Print ("Division:", num1 / num2)
```

Result: The program successfully performed all arithmetic operation on the give input and displayed the results.

23/7/2025

a) Evaluate Relational expressions.

Aim:- to develop a python program that compares two numeric values using relational operators, and display the result of each comparison.

Algorithm :-

- 1) start the program
2. Accept the two numbers from the user
3. Apply the following relational operation
 - 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)
```

outputs: first message waiting + action,

Enter the first score 1.85

Enter the second score 1.90

abc : false

acb : true

now sets work elements out
a == b : false

abc : true

acb : false

ac = b : true,

```
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 for the two given inputs

Date: 23/07/2025

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:-~~

```
test1 = int(input("Enter marks for test1"))
test2 = int(input("Enter marks for test2"))
test3 = int(input("Enter marks for test3"))
```

current:

Pass tests for 3 : 38 in about
1000 marks from test 1 to 1000
of about 1000 lines of code
takes marks for test 3 : 38

takes marks for test 3 : 42

Passed all test : false

Passed at least one test (true) with
marks for all marks since

Failed all tests : false -

status of ~~current~~ has good

(breakfast) the break conditions are
now tested for break 3 : 38 : 38 : 38

(100) about the break 3 : 38 : 38 : 38

about

the

break

Print ("Passed all tests" test1 > 40 and test2 > 40
and test3 > 40)

Print ("Passed all tests, one test," test1 > 40
or test2 > 40 or test3 > 40)

Print ("Failed all tests:" not (test1 > 40 or
test2 > 40 or test3 > 40))

VEL TECH - CSE	
EX NO.	1
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	1
TOTAL (20)	15
SIGN WITH DATE	26/12/2023

Result:- The program
logical expressions
identified Pass (fail)
on test 3
efficiently evaluated
and correctly
condition based