****

**Department of Computing Technologies**

21CSS101J PROGRAMMING FOR PROBLEM-SOLVING

**LLJ2 – REFUTE QUESTIONS**

**A REFUTE QUESTION REPORT – PYTHON**

**Submitted by**

**PARKAVI G**

**Registration Number: RA2411030010005**

**Section: u1**

**Problem Statement 1:**

The task is to Write a program to perform arithmetic operations in the python Programming language.

**Solution code:**

num1 = int(input(" Please Enter the First Value Number 1: "))

num2 = int(input(" Please Enter the Second Value Number 2: "))

# Add Two Numbers

add = num1 + num2

# Subtracting num2 from num1

sub = num1 - num2

# Multiply num1 with num2

multi = num1 \* num2

# Divide num1 by num2

div = num1 / num2

# Modulus of num1 and num2

mod = num1 % num2

# Exponent of num1 and num2

expo = num1 \*\* num2

print("The Sum of num1 and num2 is ", add)

print("The Subtraction of num1 and num2 is", sub)

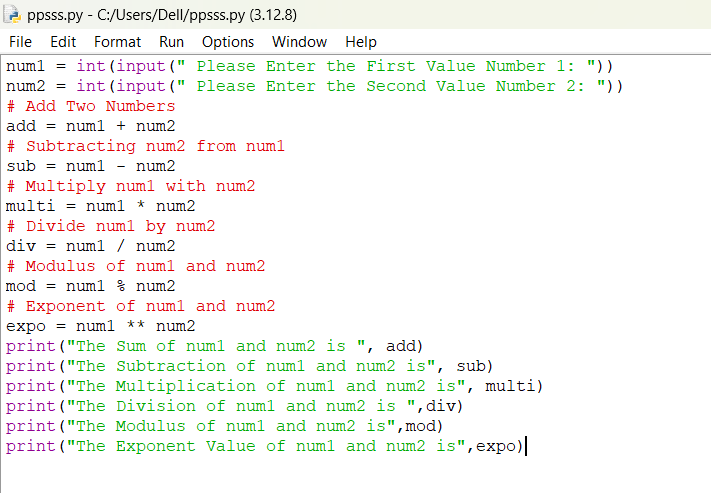
print("The Multiplication of num1 and num2 is", multi)

print("The Division of num1 and num2 is ",div)

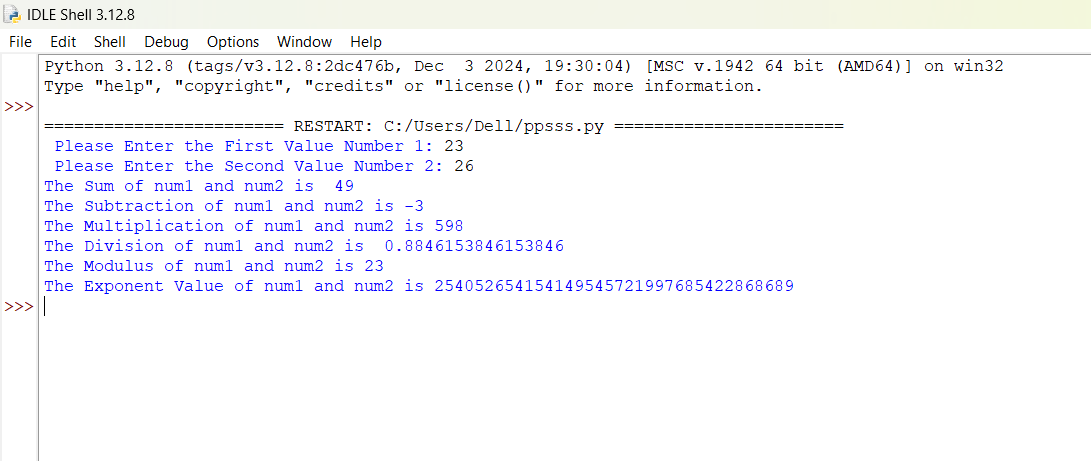
print("The Modulus of num1 and num2 is",mod)

print("The Exponent Value of num1 and num2 is",expo)

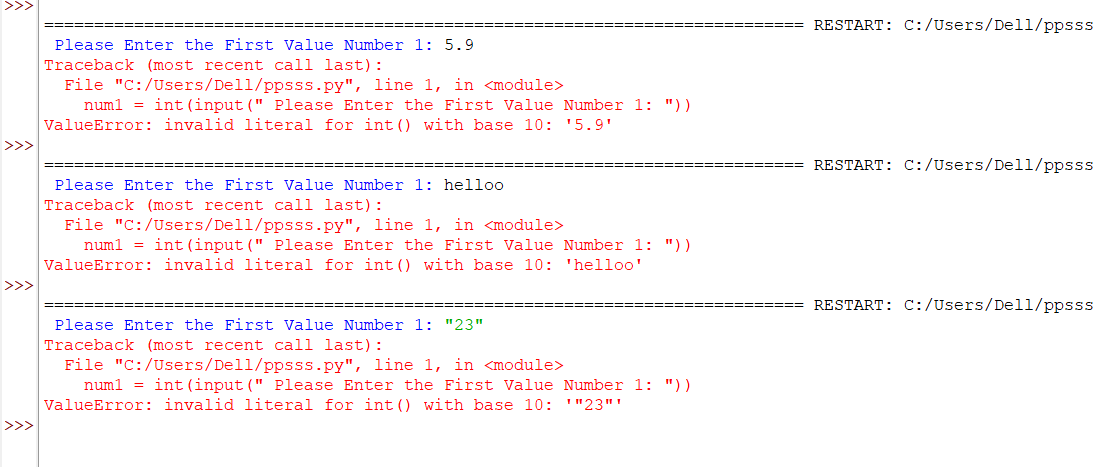
**Execution:**



*Output:*



**Refuse Test cases:**



**Explanation:**

The program will refuse input if the user enters a value that cannot be converted to an integer. For example, the following types of inputs will trigger a ValueError:

* String inputs: Any non-numeric text (e.g., "hello", "abc", "123abc")
* Floating-point numbers: Any decimal values (e.g., 3.14, 0.5)
* Special characters: Any symbols or characters (e.g., @, #, \*)
* Empty input: If the user presses Enter without typing anything.
* **Non-integer values**: If the user inputs anything that is not a valid integer (e.g., a string, float, or empty input), the program will refuse to proceed and print an error message about invalid input.
* **Zero as the second number**: If the user enters 0 for the second number, the program will refuse to perform division or modulus and will print an error message about division by zero.

**Problem statement 2:**

The task is to write a Python Program to Find the Square Root

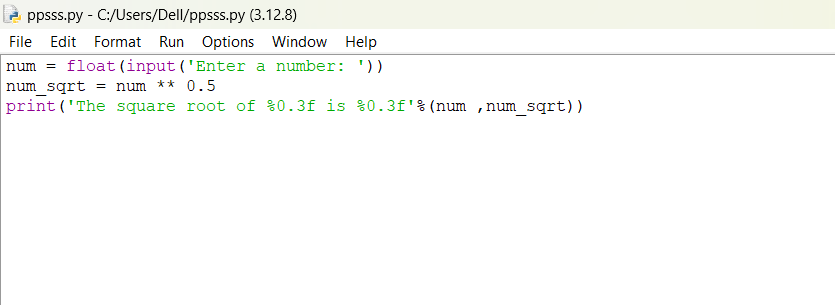
**Solution code:**

num = float (input ('Enter a number: '))

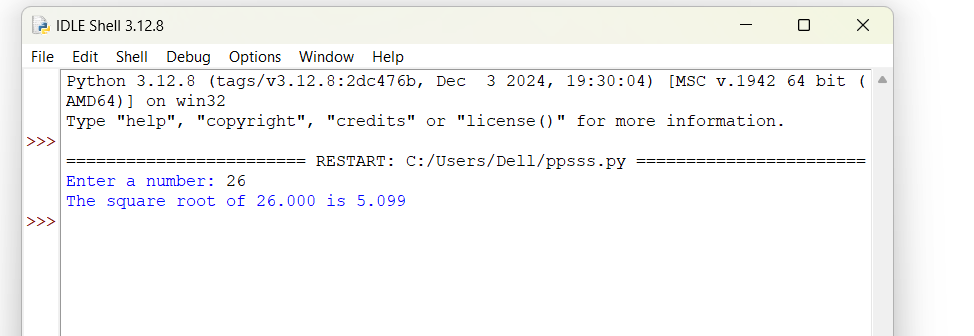
num\_sqrt = num \*\* 0.5

print ('The square root of %0.3f is %0.3f'% (num ,num\_sqrt))

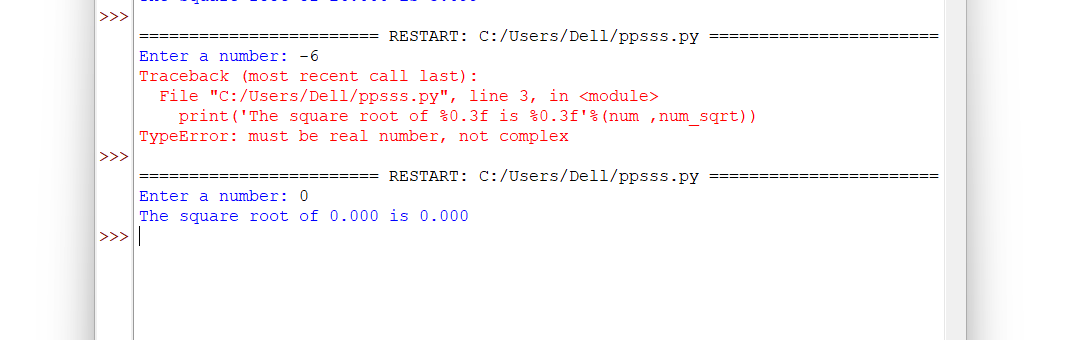
**Execution:**



*Output:*



**Refuse Test cases:**



**Explanation:**

**1. Negative numbers:**

If the user enters a negative number, the code will attempt to calculate its square root, which is not defined for real numbers. In Python, the \*\* 0.5 operation on a negative number results in a complex number, but since you are not handling complex numbers explicitly, the program will show an unexpected result.

**2. Zero:**

If the user enters 0, the square root of 0 is 0 (which is correct), and the program will correctly output the result. However, if you want to explicitly handle zero (or highlight it), you can add a special case.