Integer Range Validator

Write a Python script that asks the user to enter a number within a specified range (e.g., 1 to 100). Handle exceptions for invalid inputs and out-of-range numbers.

Input Format:

User inputs a number.

Output Format:

Confirm the input or print an error message if it's invalid or out of range.

Program:

```
try:
        user_input = input("Please enter a number between 1 and 100: ").strip()
        number = int(user_input)
        if 1 <= number <= 100:
                print("Valid input.")
        else:
                print("Error: Number out of allowed range.")
except ValueError:
        print("Error: invalid literal for int()")
```

For example:

Input	Result
10 2	5.0
10 0	Error: Cannot divide or modulo by zero.
ten 5	Error: Non-numeric input provided.

Ex. No. : 11.3 Date:

Register No.: Name:

Robust Division Calculator

Develop a Python program that safely performs division between two numbers provided by the user. Handle exceptions like division by zero and non-numeric inputs.

Input Format: Two lines of input, each containing a number.

Output Format: Print the result of the division or an error message if an exception occurs.

Program:

```
try:
        numerator_input = input("Please enter the numerator: ").strip()
        numerator = float(numerator_input)
        denominator_input = input("Please enter the denominator: ").strip()
        denominator = float(denominator_input)
        result = numerator / denominator
        print(f"The result of the division is {result:.2f}")
except ValueError:
        print("Error: Non-numeric input provided.")
except ZeroDivisionError:
        print("Error: Cannot divide by zero.")
```

For example:

Input	Result
16	The square root of 16.0 is 4.00
-4	Error: Cannot calculate the square root of a negative number.
rec	Error: could not convert string to float

User inputs a number.

Output Format:

Print the square root of the number or an error message if an exception occurs.

Program:

```
import math
user_input = input("Please enter a number: ").strip()
try:
        number = float(user_input)
        if number < 0:
                print("Error: Cannot calculate the square root of a negative number.")
        else:
                square_root = math.sqrt(number)
                print(f"The square root of {number:.2f} is {square_root:.2f}")
except ValueError:
        print(f"Error: could not convert string to float")
```

For example:

Input	Result
twenty	Error: Please enter a valid age.
25	You are 25 years old.
-1	Error: Please enter a valid age.

Ex. No. **:** **11.5** **Date:**

Register No.: **Name:**

Validated User Input

Write a Python program that asks the user for their age and prints a message based on the age. Ensure that the program handles cases where the input is not a valid integer.

Input Format: A single line input representing the user's age.

Output Format: Print a message based on the age or an error if the input is invalid.

Program:

```
age_input = input("Please enter your age: ").strip()


try:
    age = int(age_input)
    if age < 0:
        print("Error: Please enter a valid age.")
    else:
        print(f"You are {age} years old.")
except ValueError:
    print("Error: Please enter a valid age.")
     #print("arvijayakumar")
```

Ex. No. : 11.1 **Date:**

Register No.: **Name:**

Division and Modulo Calculator

Write a Python program that performs division and modulo operations on two numbers provided by the user. Handle division by zero and non-numeric inputs.

Input Format:

Two lines of input, each containing a number.

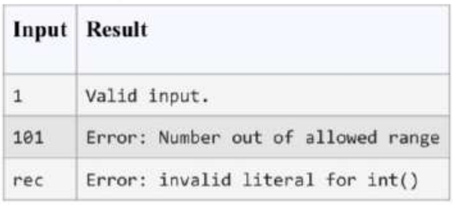
Output Format:

Print the result of division and modulo operation, or an error message if an exception occurs.

Program:

```
try:
        numerator_input = input().strip()
        numerator = float(numerator_input)
        denominator_input = input().strip()
        denominator = float(denominator_input)
        division_result = numerator / denominator
        print(f"Division result: {division_result}")
        modulo_result = numerator % denominator
        print(f"Modulo result: {modulo_result}")
except ValueError:
        print("Error: Non-numeric input provided.")
except ZeroDivisionError:
        print("Error: Cannot divide or modulo by zero.")
```

For example:



Input	Result
1	Valid input.
101	Error: Number out of allowed range
rec	Error: invalid literal for int()