WEEK 11:

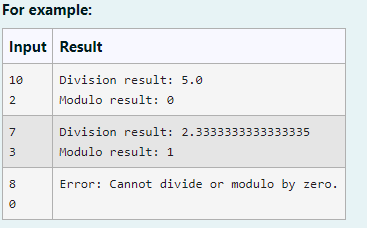
1. Write a Python program that performs division and modulo operations on two numbers provided by the user. Handle division by zero and non-numeric inputs.

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:

a=int(input())

b=int(input())

print("Division result:",a/b)

print("Modulo result:",a%b)

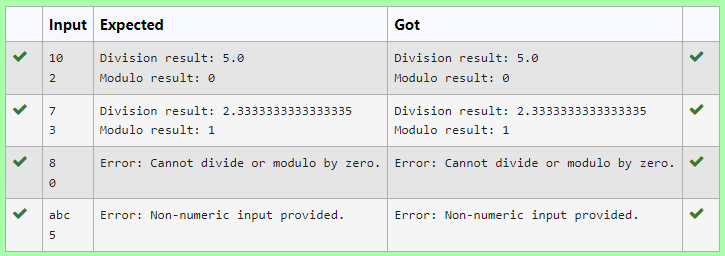
except ZeroDivisionError:

print("Error: Cannot divide or modulo by zero.")

except:

print("Error: Non-numeric input provided.")

**Output:**

****

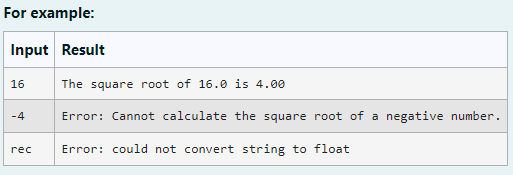
2. Develop a Python program that safely calculates the square root of a number provided by the user. Handle exceptions for negative inputs and non-numeric inputs.

Input Format:

User inputs a number.

Output Format:

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



**Program:**

import math

try:

user\_input = input()

number = float(user\_input)

if number < 0:

print("Error: Cannot calculate the square root of a negative number.")

else:

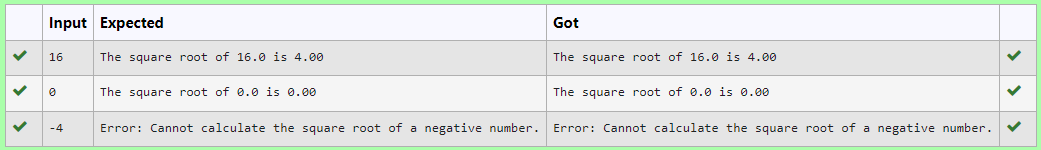
sqrt\_result = math.sqrt(number)

print(f"The square root of {number} is {sqrt\_result:.2f}")

except ValueError:

print("Error: could not convert string to float")

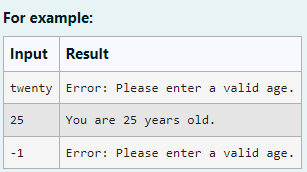
**Output:**

****

3. 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:**

try:

a=input()

if int(a)>=0:

print("You are", a,"years old.")

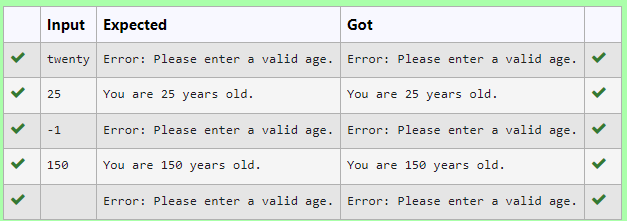
else:

print("Error: Please enter a valid age.")

except:

print("Error: Please enter a valid age.")

**Output:**

****

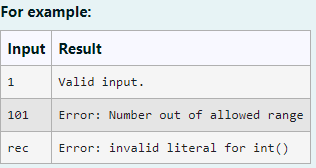
4. Write a Python script that asks the user to enter a number within a specified range (e.g., 1 to 100). Handle exceptions for invalid inputs and out-of-range numbers.

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:

a=input()

if int(a)<1 or int(a)>100:

print("Error: Number out of allowed range")

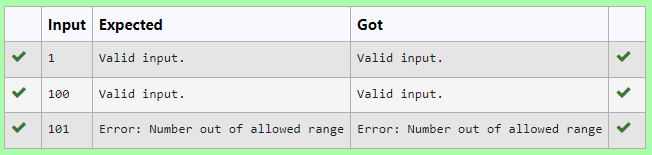
else:

print("Valid input.")

except:

print("Error: invalid literal for int()")

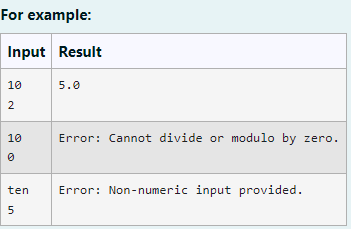
**Output:**

****

5. 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:

a=input()

b=input()

a=float(a)

b=float(b)

print(a/b)

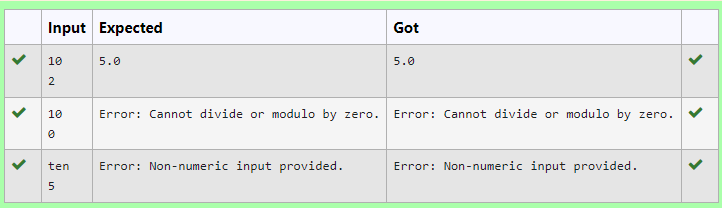
except ZeroDivisionError:

print("Error: Cannot divide or modulo by zero.")

except ValueError:

print("Error: Non-numeric input provided.")

**Output:**

****