**11 – Exception Handling**

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

**Register No.: 231801024 Name: Boopesh.R**

**EXCEPTION HANDLING**

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.

**For example:**

| **Input** | **Result** |
| --- | --- |
| 10  2 | Division result: 5.0  Modulo result: 0 |
| 7  3 | Division result: 2.3333333333333335  Modulo result: 1 |
| 8  0 | Error: Cannot divide or modulo by zero. |

**PROGRAM:**

try:

num1 = input().strip()

num2 = input().strip()

num1 = float(num1)

num2 = float(num2)

division\_result = num1 / num2

modulo\_result = int(num1% num2)

print(f"Division result: {division\_result}")

print(f"Modulo result: {modulo\_result}")

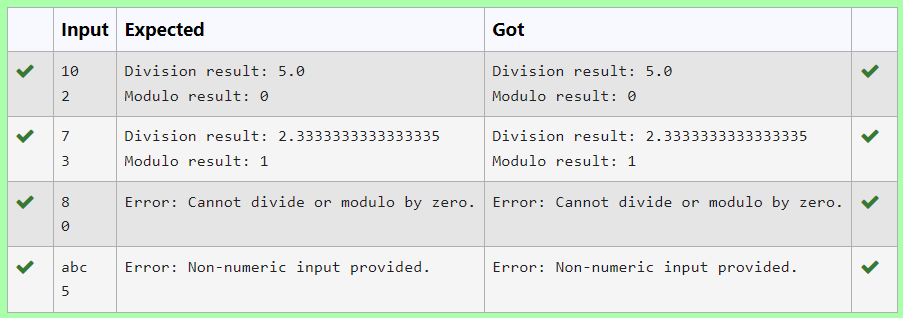
except ValueError:

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

except ZeroDivisionError:

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

**OUTPUT:**



**Ex. No. : 11.2 Date:**

**Register No.: 231801024 Name: Boopesh.R**

**EXCEPTION HANDLING**

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.

**For example:**

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

**PROGRAM:**

try:

a=input()

if len(a)==0:

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

elif a.isnumeric():

print (f"You are {a} years old.")

else:

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

except:

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

**OUTPUT:**

****

**Ex. No. : 11.3 Date:**

**Register No.: 231801024 Name: Boopesh.R**

**EXCEPTION HANDLING**

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.

**For example:**

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

**PROGRAM:**

try:

a=input()

if len(a)==0:

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

elif a.isnumeric():

print (f"You are {a} years old.")

else:

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

except:

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

**OUTPUT:**

****

**Ex. No. : 11.4 Date:**

**Register No.: 231801024 Name: Boopesh.R**

**EXCEPTION HANDLING**

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.

**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.

**For example:**

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

**PROGRAM:**

try:

a=input()

if len(a)==0:

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

elif a.isnumeric():

print (f"You are {a} years old.")

else:

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

except:

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

**OUTPUT:**

****

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

**Register No.: 231801024 Name: Boopesh.R**

**EXCEPTION HANDLING**

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:**

import math

try:

a=float(input())

if a>=0:

b=a\*\*0.5

c="%.2f"%b

print("The square root of",float(a),"is",c)

else:

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

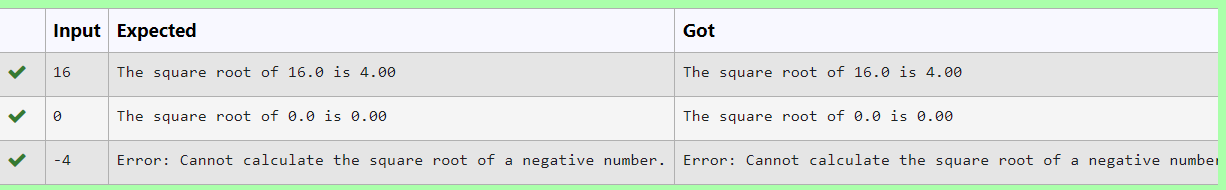
except EOFError:

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

except ValueError:

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

**OUTPUT:**

****