

•EXCEPTIONS

Ex. No. : 11.1

Date: 1/6/2024

Register No: 231501057

Name: HARISH KUMAR V

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

PROGRAM:

try:

```
a = int(input())
b = int(input())
print(f'Division result: {a/b}\nModulo result: {a%b}')
```

except ZeroDivisionError:

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

except ValueError:

```
    print('Error: Non-numeric input provided.')
```

	Input	Expected	Got
✓	10 2	Division result: 5.0 Modulo result: 0	Division result: 5.0 Modulo result: 0
✓	7 3	Division result: 2.3333333333333335 Modulo result: 1	Division result: 2.333333333333333 Modulo result: 1
✓	8 0	Error: Cannot divide or modulo by zero.	Error: Cannot divide or modulo by
✓	abc 5	Error: Non-numeric input provided.	Error: Non-numeric input provided

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Ex. No. : 11.2

Date: 1/6/2024

Register No: 231501057

Name: HARISH KUMAR

V

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.

For example:

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

PROGRAM:

try:

```
user_input = int(input(""))
if 1 <= user_input <= 100:
    print("Valid input.")
else:
    print("Error: Number out of allowed range")
except ValueError:
    print("Error: invalid literal for int()")
```

	Input	Expected	Got	
✓	1	Valid input.	Valid input.	✓
✓	100	Valid input.	Valid input.	✓
✓	101	Error: Number out of allowed range	Error: Number out of allowed range	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Ex. No. : 11.3

Date: 1/6/2024

Register No: 231501057

Name: HARISH KUMAR

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.

For example:

For example:

Input	Result
10 2	Division result: 5.0 Modulo result: 0
7 3	Division result: 2.333333333333335 Modulo result: 1
8 0	Error: Cannot divide or modulo by zero.

PROGRAM:

```
while True:  
    try:  
        num1 = float(input(""))  
        num2 = float(input(""))  
  
        result = num1 / num2  
        print(f'{result}')  
        break  
  
    except ValueError:  
        print("Error: Non-numeric input provided.")  
        break  
  
    except ZeroDivisionError:  
        print("Error: Cannot divide or modulo by zero.")  
        break
```

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

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Ex. No. : 11.4

Date: 1/6/2024

Register No: 231501057

Name: HARISH KUMAR V

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 = int(input())
```

```
if a<0:
```

```
    print('Error: Please enter a valid age.')
```

```
else:
```

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

```
except (ValueError, EOFError):
```

```
    print('Error: Please enter a valid age.')
```

OUTPUT:

	Input	Expected	Got	
✓	twenty	Error: Please enter a valid age.	Error: Please enter a valid age.	✓
✓	25	You are 25 years old.	You are 25 years old.	✓
✓	-1	Error: Please enter a valid age.	Error: Please enter a valid age.	✓
✓	150	You are 150 years old.	You are 150 years old.	✓
✓		Error: Please enter a valid age.	Error: Please enter a valid age.	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Ex. No. : 11.5

Date: 1/6/2024

Register No: 231501057

Name: HARISH KUMAR V

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.

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

PROGRAM:

```
import math

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

OJUTPUT:

	Input	Expected	Got
✓	16	The square root of 16.0 is 4.00	The square
✓	0	The square root of 0.0 is 0.00	The square
✓	-4	Error: Cannot calculate the square root of a negative number.	Error: Cann

Passed all tests! ✓

Correct
Marks for this submission: 1.00/1.00.