

# Python Programming

## Lab :- 11(Exception)

**Student Id:- AF0417098**

**Student Name:- Ankush**

1. Write a Python program to handle a ZeroDivisionError exception when dividing a number by zero.

**Program :-**

```
labassignment11.py > ...
1  def divide_numbers(a, b):
2      try:
3          result = a / b
4      except ZeroDivisionError:
5          return "Error: Cannot divide by zero."
6      else:
7          return result
8
9  # Example usage:
10 num1 = float(input("Enter the numerator: "))
11 num2 = float(input("Enter the denominator: "))
12 print(divide_numbers(num1, num2))
13
```

**Output:-**

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  TEST RESULTS  PORTS
PS C:\Users\Raj Kumar\Desktop\python programming> & "C:/Users/Raj Kumar/AppData/Local/Programs/Python/Python312/python.exe" "c:/Users/Raj Kumar/Desktop/python programming/labassignment11.py"
Enter the numerator: 12
Enter the denominator: 6
2.0
PS C:\Users\Raj Kumar\Desktop\python programming> & "C:/Users/Raj Kumar/AppData/Local/Programs/Python/Python312/python.exe" "c:/Users/Raj Kumar/Desktop/python programming/labassignment11.py"
Enter the numerator: 3
Enter the denominator: 0
Error: Cannot divide by zero.
PS C:\Users\Raj Kumar\Desktop\python programming>
```

2. Write a Python program that prompts the user to input an integer and raises a ValueError exception if the input is not a valid integer.

**Program:-**

```
labassignment10.py labassignment9.py labassignment11.py X
labassignment11.py > ...
11 # num2 = float(input("Enter the denominator: "))
12 # print(divide_numbers(num1, num2))
13
14 def get_integer_input():
15     try:
16         num = int(input("Enter an integer: "))
17         return num
18     except ValueError:
19         return "Error: That is not a valid integer."
20
21 # Example usage:
22 print(get_integer_input())
23
```

Output:-

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS
PS C:\Users\Raj Kumar\Desktop\python programming> & "C:/Users/Raj Kumar/AppData/Local/Programs/Python/Python312/pytho
n.exe" "c:/Users/Raj Kumar/Desktop/python programming/labassignment11.py"
Enter an integer: 23
23
PS C:\Users\Raj Kumar\Desktop\python programming> & "C:/Users/Raj Kumar/AppData/Local/Programs/Python/Python312/pytho
n.exe" "c:/Users/Raj Kumar/Desktop/python programming/labassignment11.py"
Enter an integer: ankush
Error: That is not a valid integer.
PS C:\Users\Raj Kumar\Desktop\python programming>
```

3. Write a Python program that opens a file and handles a `FileNotFoundError` exception if the file does not exist.

Program:-

```
labassignment10.py X labassignment9.py labassignment11.py X story.txt
labassignment11.py > ...
24
25 def open_file(filename):
26     try:
27         with open(filename, 'r') as file:
28             content = file.read()
29             return content
30     except FileNotFoundError:
31         return "Error: File not found."
32
33 # Example usage:
34 filename = input("Enter the filename: ")
35 print(open_file(filename))
36
37
```

Output:-

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS
PS C:\Users\Raj Kumar\Desktop\python programming> & "C:/Users/Raj Kumar/AppData/Local/Programs/Python/Python312/python.exe" "c:/Users/Raj Kumar/Desktop/python programming/labassignment11.py"
Enter the filename: story.txt
This is a sample text file.
It contains multiple lines.
PS C:\Users\Raj Kumar\Desktop\python programming> & "C:/Users/Raj Kumar/AppData/Local/Programs/Python/Python312/python.exe" "c:/Users/Raj Kumar/Desktop/python programming/labassignment11.py"
Enter the filename: story1.txt
Error: File not found.
PS C:\Users\Raj Kumar\Desktop\python programming> |
```

**4. Write a Python program that prompts the user to input two numbers and raises a `TypeError` exception if the inputs are not numerical**

**Program:-**

```
labassignment10.py x labassignment9.py labassignment11.py x story.txt
labassignment11.py > ...
35 # print(open_file(filename))
36
37
38 def get_two_numbers():
39     try:
40         num1 = float(input("Enter the first number: "))
41         num2 = float(input("Enter the second number: "))
42         return num1, num2
43     except ValueError:
44         raise TypeError("Error: Inputs must be numerical.")
45
46 # Example usage:
47 try:
48     num1, num2 = get_two_numbers()
49     print(f"The numbers you entered are {num1} and {num2}.")
50 except TypeError as e:
51     print(e)
```

**Output:-**

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS
PS C:\Users\Raj Kumar\Desktop\python programming> & "C:/Users/Raj Kumar/AppData/Local/Programs/Python/Python312/python.exe" "c:/Users/Raj Kumar/Desktop/python programming/labassignment11.py"
Enter the first number: ank
Error: Inputs must be numerical.
PS C:\Users\Raj Kumar\Desktop\python programming> & "C:/Users/Raj Kumar/AppData/Local/Programs/Python/Python312/python.exe" "c:/Users/Raj Kumar/Desktop/python programming/labassignment11.py"
Enter the first number: 12
Enter the second number: 13
The numbers you entered are 12.0 and 13.0.
PS C:\Users\Raj Kumar\Desktop\python programming> |
```