**Object-Oriented Programming Lab Report**

**Autumn 2023**

**Lab Experiment 7**

Name: Ansh Garg

Roll no.: R2142220030

Sap\_ID: 500105940

B. Tech. CSE spz. Fullstack AI, Semester 2, Batch 3:

Date of experiment: 29-09-2023



School of Computer Science,

University of Petroleum and Energy Studies,

Dehradun

**Experiment 1**

**Assignment-1**

*Write a program to demonstrate exception handling.*

**Java code:**

// wap a java to demonstarate exception handling

// Author: Ansh Garg

class exp {

    public static float divide(int a, int b) {

        float c= a/b;

        return c;

    }

    public static void main(String[] args) {

        int a=1;

        int b=0;

        try {

            System.out.println("TRY Block entered");

            float d = divide(a,b);

            System.out.println("Division is: "+d);

        } catch (ArithmeticException ex) {

            System.out.println("CATCH Block entered");

            System.out.println(ex.getMessage());

        } finally {

            System.out.println("Divide Function Calling Done");

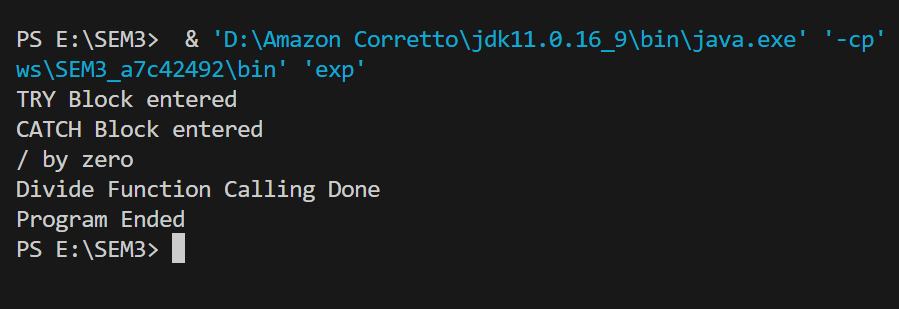
        }

        System.out.println("Program Ended");

    }

}

**Screenshot of the output:**



**Assignment-2**

*WAP in Java*

*1)wap a program to handle another type of exception according to your choice*

*2)inlcude user input, this should create an exception*

*3)print the exception message to show that expetion is caught*

*4)Identify the exception handling keywords used in the program and why*

*5)Create a customised exception class for same code*

**Java Code:**

// 1)wap a program to handle another type of exception according to your choice

// 2)inlcude user input, this should create an exception

// 3)print the exception message to show that expetion is caught

// 4)Identify the exception handling keywords used in the program and why

// 5)Create a customised exception class for same code

import java.util.Scanner;

class MyException extends Exception {

    public MyException(String errorMessage) {

        super(errorMessage);

    }

}

class AnotherException extends Exception {

    public AnotherException(String errorMessage) {

        super(errorMessage);

    }

}

public class Assignment {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter a number: ");

        int num = sc.nextInt();

        try {

            if (num < 0) {

                throw new MyException("Number cannot be negative");

            } else if (num > 100) {

                throw new AnotherException("Number cannot be greater than 100");

            }

        } catch (MyException e) {

            System.out.println("MyException caught: " + e.getMessage());

        } catch (AnotherException e) {

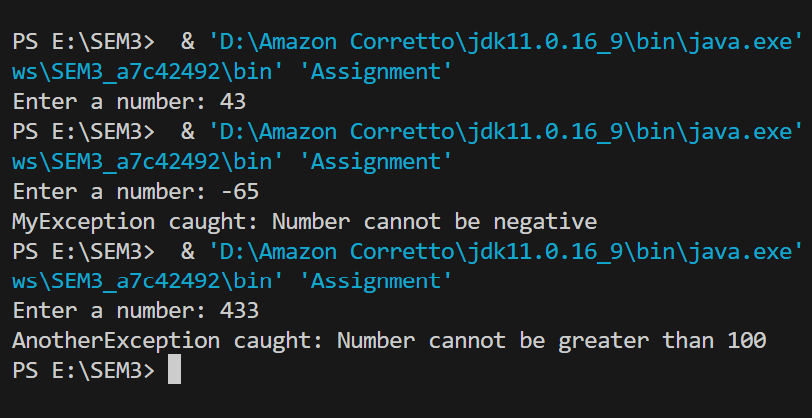
            System.out.println("AnotherException caught: " + e.getMessage());

        }

    }

}

**Output:**

****