

Aashish Shrestha

Unit #1.3

1) Write a program to demonstrate try-catch-finally.

Program:

```
package pkg1.pkg3.pkg1;

import java.util.Scanner;

/**
 *
 * @author aashish
 */
public class Main {

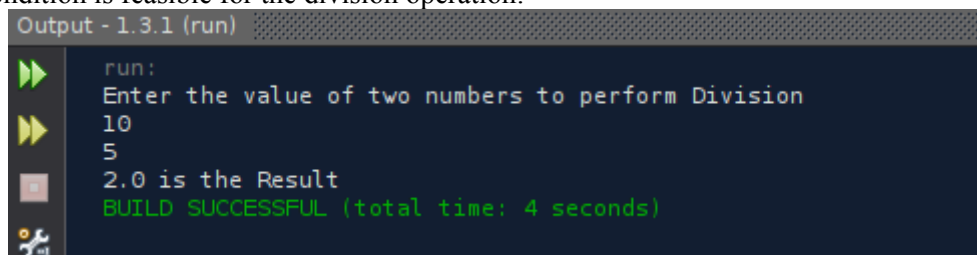
    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int a,b;
        double c;
        Scanner input = new Scanner(System.in);
        System.out.println("Enter the value of two numbers to perform
Division");
        a = input.nextInt();
        b = input.nextInt();
        try{
            c = a/b;
            System.out.println(c + " is the Result");
        }
        catch(Exception e){
            System.out.println("Dividing operation Could't be performed");
        }

    }

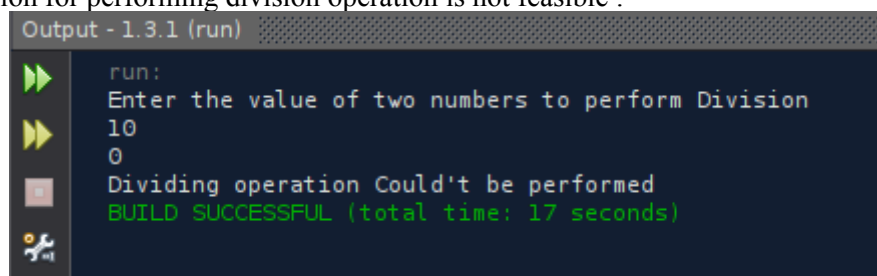
}
```

Output:

When the condition is feasible for the division operation:

A screenshot of an IDE's output window titled "Output - 1.3.1 (run)". It shows the execution of a Java program. The output text is: "run: Enter the value of two numbers to perform Division", followed by the user input "10" and "5", then "2.0 is the Result", and finally "BUILD SUCCESSFUL (total time: 4 seconds)". On the left side of the output window, there are icons for running (a green play button), stepping through (a yellow play button), and debugging (a red bug icon).

When the condition for performing division operation is not feasible :

A screenshot of an IDE's output window titled "Output - 1.3.1 (run)". It shows the execution of the same Java program. The output text is: "run: Enter the value of two numbers to perform Division", followed by the user input "10" and "0", then "Dividing operation Could't be performed", and finally "BUILD SUCCESSFUL (total time: 17 seconds)". On the left side of the output window, there are icons for running (a green play button), stepping through (a yellow play button), and debugging (a red bug icon).