

Name: Pratyay Kumar
Aggie Id: 800811773
Date: 10/05/2022
Lab Assignment: 5

Program #5 Exception Handling in JAVA

Problem Description: Task is to rewrite the grade distribution ADA program in JAVA. In the JAVA version, we must change the second half of the first loop so that all assignments to the counting Array "Freq ()" are updated in the Exception portion of the code. There should be no valid updates to "Freq ()" anywhere else in the loop.

Rewritten code in JAVA:

```
/*
 * Name: Pratyay Kumar
 * Date: 09/02/2022
 * Lab: 5
 * Purpose:
 * Program Description: Task, in this JAVA version, is to change the second half of
the first loop so that all assignments to the counting Array "Freq[]" are updated in
the Exception portion of the code. There should be no valid updates to "Freq[]"
anywhere else in the loop.
*/
import java.util.*;

class lab5 {
    public static void main (String[] args) {

        // Creating a scanner object
        Scanner s = new Scanner (System.in);

        // Declared the required variable and array here.
        int[] freq = new int[11]; // The array has been declared of size 11.
        int limit_1, limit_2, index;
        int new_grade;

        /* Getting into the loop.
        * The loop will run unlimited times unless there is a negative number in the
input.
        */
        for ( ;; ) {
            // Scanning input
            new_grade = s.nextInt();
```

```

        // This try and catch will help to break from loop, whenever a negative
        number is inputted.
        try {
            if (new_grade < 0) {
                throw new ArithmeticException();
            }
        } catch (ArithmeticException e) {
            break;
        }

        // Scaling index
        index = new_grade/10 + 1;

        // Taking every input grade as an exception.
        try {
            if (new_grade >= 0) {
                throw new ArithmeticException();
            }
        }
        // As part of the logic the freq[] updating is done in the catch section.
        catch (ArithmeticException e) {
            if (new_grade < 100)
                freq[index] = freq[index] + 1;
            if (new_grade == 100)
                freq[10] = freq[10] + 1;
            if (new_grade > 100)
                System.out.println("Error -- new grade: " + new_grade + " is out
of range");
        }

    }

    System.out.println("Limits Frequency");
    System.out.println("");

    // Printing out the values.
    // We can reach 90-100 index.
    for ( index=0; index<10; index++ ) {
        limit_1 = 10 * index;
        limit_2 = limit_1 + 9;
        if (index == 9)
            limit_2 = 100;
        System.out.print(limit_1 + " ");
        System.out.print(limit_2 + " ");
        System.out.print(freq[index+1]);
        System.out.println("");
    }

```

```
}  
}
```

Output for the above code:

```
.../Desktop/Fall 2022/CS 471 Programing Language Structures 1/Assignment/5 /usr/bin/env /Library/Internet\ Plug-Ins/JavaAppletPlugin.plugin/Contents/Home/bin/java -cp /Users/pratyaykumar/Library  
/Application\ Support/Code/User/workspaceStorage/39fe4ad9417207b22de0f088ef2ebf68/redhat.java/jdt_ws/5_744172c9/bin lab5  
10  
11  
12  
111  
Error -- new grade: 111 is out of range  
100  
99  
0  
2  
3  
4  
-1  
Limits Frequency  
0 9 4  
10 19 3  
20 29 0  
30 39 0  
40 49 0  
50 59 0  
60 69 0  
70 79 0  
80 89 0  
90 100 2  
.../De/Fall 2022/CS 471 Programing Language Structures 1/Assignment/5 21s 07:01:47 PM
```

The above output shows the program can take any amount of positive number numbers, and it terminates whenever a negative number comes.

The above output also gives the frequency of number in the range of 90-100.

Program description:

The array (freq) has declared of size 11, so it can have 1 to 10 as indexes. I am using try catch to exception handling the cases whenever the index is out of bound or almost out of bound. Taking everything as an exception, I have put all the operations done on the freq [] array into the catch section of the code with some logic.