Question **1**

Correct

Marked out of 5.00

Write a Java program to handle ArithmeticException and ArrayIndexOutOfBoundsException.

Create an array, read the input from the user, and store it in the array.

Divide the 0th index element by the 1st index element and store it.

if the 1st element is zero, it will throw an exception.

if you try to access an element beyond the array limit throws an exception.

Input:

5

10 0 20 30 40

Output:

java.lang.ArithmeticException: / by zero
I am always executed

Input:

3
10 20 30

Output
java.lang.ArrayIndexOutOfBoundsException: Index 3 out of bounds for length 3
I am always executed

For example:

Test	Input	Result
1	6 1 0 4 1 2 8	java.lang.ArithmeticException: / by zero I am always executed

Answer: (penalty regime: 0 %)

```
import java.util.Scanner;

public class ExceptionHandlingExample {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        try {
            int n = scanner.nextInt();

            // Initialize the array
            int[] array = new int[n];

            for (int i = 0; i < n; i++) {
                array[i] = scanner.nextInt();
            }

            // Try dividing the element at index 0 by the element at index 1
            int result = array[0] / array[1]; // Potential ArithmeticException

           // System.out.println("Division result: " + result);

            // Try accessing an element beyond the array bounds
            //System.out.println("Accessing element at index " + n);
            System.out.println("Element: " + array[n]); // Potential ArrayIndexOutOfBoundsE

        } catch (ArithmeticException e) {
            // Handle division by zero
            System.out.println(e.toString());
        } catch (ArrayIndexOutOfBoundsException e) {
            // Handle accessing an out-of-bound array index
            System.out.println(e.toString());
        } finally {
            // This block is always executed
            System.out.println("I am always executed");
        }

        // Close the scanner
        scanner.close();
    }
}
```

```
42
```

	Test	Input	Expected	Got
✓	1	6 1 0 4 1 2 8	java.lang.ArithmeticException: / by zero I am always executed	java.lang.ArithmeticException: / by zero I am always executed
✓	2	3 10 20 30	java.lang.ArrayIndexOutOfBoundsException: Index 3 out of bounds for length 3 I am always executed	java.lang.ArrayIndexOutOfBoundsException: Index 3 out of bounds for length 3 I am always executed

Passed all tests! ✓

Question **2**

Correct

Marked out of 5.00

In the following program, an array of integer data is to be initialized.
During the initialization, if a user enters a value other than an integer, it will throw an InputMismatchException exception.
On the occurrence of such an exception, your program should print “You entered bad data.”
If there is no such exception it will print the total sum of the array.

/* Define try-catch block to save user input in the array "name"
If there is an exception then catch the exception otherwise print the total sum of the array. */

Sample Input:

3
5 2 1

Sample Output:

8

Sample Input:

2

1 g

Sample Output:

```
You entered bad data.
```

For example:

Input	Result
3 5 2 1	8
2 1 g	You entered bad data.

Answer: (penalty regime: 0 %)

Reset answer

```java
import java.util.Scanner;
import java.util.InputMismatchException;

public class Prog {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int length = sc.nextInt();

        // Create an array to save user input
        int[] name = new int[length];
        int sum = 0; // Save the total sum of the array

        // Define try-catch block to save user input in the array "name"
        try {
            for (int i = 0; i < length; i++) {
                name[i] = sc.nextInt(); // User input
                sum += name[i]; // Add to total sum
            }
            // Print the total sum of the array
            System.out.println(sum);
        } catch (InputMismatchException e) {
            System.out.println("You entered bad data.");
        } finally {
            sc.close(); // Close the scanner to prevent resource leaks
        }
    }
}

```

	Input	Expected	Got	
✓	3 5 2 1	8	8	✓
✓	2 1 g	You entered bad data.	You entered bad data.	✓

Passed all tests! ✓

Question **3**

Correct

Marked out of 5.00

Write a Java program to create a method that takes an integer as a parameter

and throws an exception if the number is odd.

Sample input and Output:

```
82 is even.
Error: 37 is odd.
```

Fill the preloaded answer to get the expected output.

For example:

Result
82 is even. Error: 37 is odd.

Answer: (penalty regime: 0 %)

Reset answer

```java
public class Prog {
    public static void main(String[] args) {
        int n1 = 82;
        int n2 = 37;

        // Call the method for the first number
        try {
            trynumber(n1);
        } catch (Exception e) {
            System.out.println(e.getMessage());
        }

        // Call the method for the second number
        try {
            trynumber(n2);
        } catch (Exception e) {
            System.out.println(e.getMessage());
        }
    }

    public static void trynumber(int n) throws Exception {
        checkEvenNumber(n); // Check if the number is even
        System.out.println(n + " is even.");
    }

    public static void checkEvenNumber(int number) throws Exception {
        if (number % 2 != 0) {
            throw new Exception("Error: " + number + " is odd.");
        }
    }
}
```

	Expected	Got	
✓	82 is even. Error: 37 is odd.	82 is even. Error: 37 is odd.	✓

Passed all tests! ✓

◄ Lab-09-MCQ

Jump to...

The "Nambiar Number" Generator ►