Status	Finished
Started	Wednesday, 9 October 2024, 2:42 PM
Completed	Wednesday, 9 October 2024, 2:44 PM
Duration	1 min 28 secs

Question **1**

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

        try {

            int n = scanner.nextInt();  // Read the size of the array

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

            int sum = 0;
            for (int i = 0; i < n; i++) {
                array[i] = scanner.nextInt();  // Read integers into the array
                sum += array[i];  // Calculate the sum
            }

            // If no exception occurs, print the total sum
            System.out.println(sum);
        } catch (InputMismatchException e) {
            // Handle the exception if non-integer input is entered
            System.out.println("You entered bad data.");
        } finally {
            scanner.close();  // Close the scanner
        }
    }
}
```

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

Passed all tests! ✓

Question **2**

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

```
1  public class EvenOddChecker {
2
3      public static void main(String[] args) {
4          checkNumber(82);
5          checkNumber(37);
6      }
7
8      // Method to check if the number is even or odd
9      public static void checkNumber(int number) {
10         try {
11             if (isOdd(number)) {
12                 throw new IllegalArgumentException(number + " is odd.");
13             } else {
14                 System.out.println(number + " is even.");
15             }
16         } catch (IllegalArgumentException e) {
17             System.out.println("Error: " + e.getMessage());
18         }
19     }
20
21     // Method to determine if a number is odd
22     public static boolean isOdd(int number) {
23         return number % 2 != 0;
24     }
25 }
26
27
```

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

Passed all tests! ✓

Question **3**

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

        // Read the size of the array

        int size = scanner.nextInt();

        // Create an array of the specified size
        int[] array = new int[size];

        // Read elements into the array

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

        // Attempt to access the elements and perform the division
        try {
            // This will intentionally access an out-of-bounds index
            int outOfBoundsAccess = array[3];  // Example access beyond the valid range

            // Now perform the division only if the access is within bounds
            int result = array[0] / array[1];
            System.out.println("Result: " + result);
        } catch (ArithmeticException e) {
            System.out.println("java.lang.ArithmeticException: " + e.getMessage());
        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println("java.lang.ArrayIndexOutOfBoundsException: " + e.getMessage());
        }

        // Always executed message
        System.out.println("I am always executed");

        // Close the scanner
```

```
38          scanner.close();
39      }
40  }
41
```

	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! ✓

◄ Lab-09-MCQ

Jump to...

The “Nambiar Number” Generator ►