**Usman Institute of Technology**

**Department of Computer Science**

**Course Code: SE312**

**Course Title: Software Construction and Development**

**SPRING 2024**

**Lab 10**

**Fault Tolerance in Java**

**Objective:** Constructing a fault tolerant program by implementing exception handling techniques.

**Student Information**

|  |  |
| --- | --- |
| Student Name | Syed Muhammad Zaid |
| Student ID | 20B-052-SE |
| Date | 5/17/2024 |

**Assessment**

|  |  |
| --- | --- |
| Marks Obtained |  |
| Remarks |  |
| Signature |  |

**HOME TASK:**

1. Write a program that meets the following requirements:

• Creates an array with 10 randomly chosen integers.

• Prompts the user to enter the index of the array, then displays the corresponding element

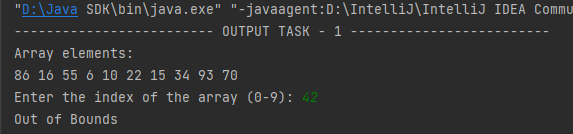
value. If the specified index is out of bounds, display the message Out of Bounds.

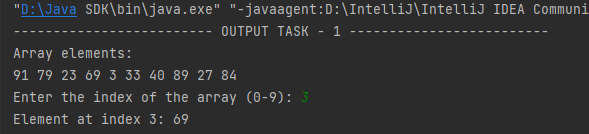
(ArrayIndexOutOfBoundsException)

**CODE:**

package com.company;  
  
import java.util.Random;  
import java.util.Scanner;  
  
public class task1 {  
 public static void main(String[] args) {  
 // Create an array with 10 randomly chosen integers  
 int[] array = new int[10];  
 Random random = new Random();  
 for (int i = 0; i < array.length; i++) {  
 array[i] = random.nextInt(100); // Random integers between 0 and 99  
 }  
  
 // Print the array for reference  
 System.*out*.println("Array elements:");  
 for (int num : array) {  
 System.*out*.print(num + " ");  
 }  
 System.*out*.println();  
  
 // Prompt the user to enter the index of the array  
 Scanner scanner = new Scanner(System.*in*);  
 System.*out*.print("Enter the index of the array (0-9): ");  
  
 try {  
 int index = scanner.nextInt();  
 // Display the corresponding element value  
 System.*out*.println("Element at index " + index + ": " + array[index]);  
 } catch (ArrayIndexOutOfBoundsException e) {  
 // Display the message if the specified index is out of bounds  
 System.*out*.println("Out of Bounds");  
 } catch (Exception e) {  
 // Handle any other exceptions  
 System.*out*.println("Invalid input. Please enter an integer between 0 and 9.");  
 }  
  
// scanner.close();  
 }  
}

**OUTPUT:**

****

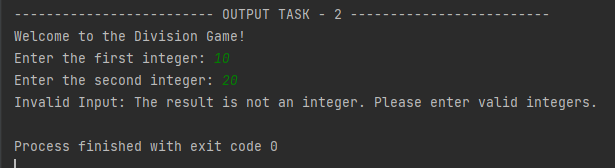
****

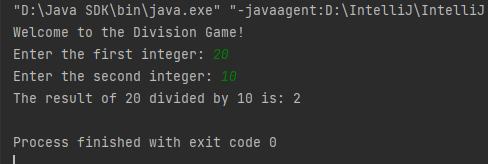
2. Suppose you are developing a game for kids in which they are learning the division operation in math. Your game will take input from kids (2 integers) and then perform the division and displays the answer. Think and apply exception handling in this scenario. For e.g. Arithmetic Exception might occur here. Also suppose, this game has the limitation that it only performs division between integers so if it gets a decimal number as input, it throws an exception stating that the input is invalid, please give integer number etc.

**CODE:**

package com.company;  
import java.util.InputMismatchException;  
import java.util.Scanner;  
  
public class task2 {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
  
 try {  
 System.*out*.println("Welcome to the Division Game!");  
 System.*out*.print("Enter the first integer: ");  
 int dividend = scanner.nextInt();  
 System.*out*.print("Enter the second integer: ");  
 int divisor = scanner.nextInt();  
  
 if (divisor == 0) {  
 throw new ArithmeticException("Division by zero is not allowed.");  
 }  
  
 if (dividend % divisor != 0) {  
 throw new IllegalArgumentException("The result is not an integer. Please enter valid integers.");  
 }  
  
 int result = dividend / divisor;  
 System.*out*.println("The result of " + dividend + " divided by " + divisor + " is: " + result);  
 } catch (InputMismatchException e) {  
 System.*out*.println("Invalid input. Please enter integers only.");  
 } catch (ArithmeticException e) {  
 System.*out*.println("Arithmetic Error: " + e.getMessage());  
 } catch (IllegalArgumentException e) {  
 System.*out*.println("Invalid Input: " + e.getMessage());  
 }  
 }  
}

**OUTPUT:**

****

****