Молдавский Государственный Университет

Факультет Математики и Информатики

Департамент Информатики

Лабораторная работа №3

по курсу Java

на тему

“Создание и использование исключений”

Выполнил: Bogdanov Iurii,

группа I2302

Проверил: Gheorghe Latul

Кишинев, 2024

**Цель работы:**

Целью данной лабораторной работы является создание собственного исключения в Java, а также использование методов throw, throws и обработки исключений с помощью try-catch. В рамках работы реализуется программа, которая запрашивает у пользователя ввод 10 чисел, проверяет их на простоту, и выводит корректные значения.

**Описание задачи:**

1. Создать собственное исключение, которое будет выбрасываться при вводе чисел, не являющихся простыми.
2. Реализовать метод для ввода чисел с использованием класса Scanner, который будет возбуждать созданное исключение, если число не является простым.
3. Обработать исключение в методе ввода, продолжая ввод чисел до достижения необходимого количества (10 чисел).
4. Вывести на экран все введенные числа, если они являются простыми, в случае успешного выполнения программы.

**Реализация:**

**1. Создание собственного исключения**

Создан класс PrimeNumberException, который наследуется от класса Exception. Он принимает неверное число в качестве аргумента и выводит сообщение об ошибке, если введенное число не является простым.

package exceptions;

public class PrimeNumberException extends Exception {  
 public PrimeNumberException(int invalidNumber) {  
 super("Ошибка: Число " + invalidNumber + " не является простым.");  
 }  
}

#### 2. Метод для ввода чисел

Создан класс PrimeNumberInput, который содержит метод inputPrimeNumbers(). Этот метод запрашивает у пользователя ввод чисел и проверяет, является ли каждое число простым. В случае неправильного ввода выбрасывается исключение PrimeNumberException.

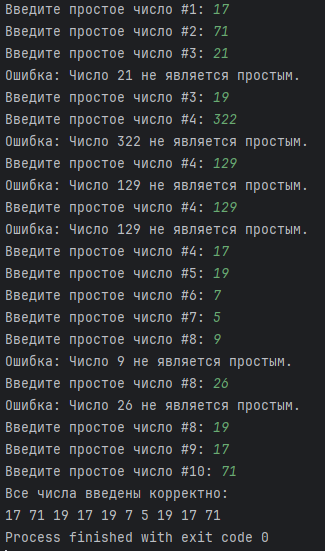
package input;  
  
import exceptions.PrimeNumberException;  
import java.util.Scanner;  
  
public class PrimeNumberInput {  
  
 private boolean isPrime(int number) throws PrimeNumberException {  
 if (number <= 1) {  
 throw new PrimeNumberException(number);  
 }  
 for (int i = 2; i <= Math.*sqrt*(number); i++) {  
 if (number % i == 0) {  
 throw new PrimeNumberException(number);  
 }  
 }  
 return true;  
 }  
  
 public int[] inputPrimeNumbers() {  
 Scanner scanner = new Scanner(System.*in*);  
 int[] primeNumbers = new int[10];  
 int count = 0;  
  
 while (count < primeNumbers.length) {  
 System.*out*.print("Введите простое число #" + (count + 1) + ": ");  
 int num = scanner.nextInt();  
 try {  
 isPrime(num);  
 primeNumbers[count] = num;  
 count++;  
 } catch (PrimeNumberException e) {  
 System.*out*.println(e.getMessage());  
 }  
 }  
 return primeNumbers;  
 }  
}

#### 3. Главный метод

В классе Main реализован основной метод программы, который вызывает метод для ввода чисел и выводит результаты на экран.

//Лабораторная работа №3. Создание и обработка собственных исключений.  
//В программе требуется:  
//Создать собственное исключение (class).  
//Создать метод (throw), который может возбуждать это исключение(throws).  
//Написать метод, перехватывающий и обрабатывающий исключение (try-catch), возбуждаемое другим методом.  
//Исключение: с консоли вводятся 10 чисел и записываются в массив. Добиться ввода только простых чисел.  
  
package main;  
  
import input.PrimeNumberInput;  
  
public class Main {  
 public static void main(String[] args) {  
 PrimeNumberInput numberInput = new PrimeNumberInput();  
  
 int[] primeNumbers = numberInput.inputPrimeNumbers();  
 System.*out*.println("Все числа введены корректно: ");  
 for (int num : primeNumbers) {  
 System.*out*.print(num + " ");  
 }  
 }  
}

**Результат вывода программы:**



Программа запрашивает у пользователя ввод 10 простых чисел. При вводе простого числа оно добавляется в массив, и программа продолжает работу до тех пор, пока массив не заполнится. Если вводится число, которое не является простым, выводится сообщение об ошибке, информирующее пользователя о неверном числе. После успешного завершения программы на экран выводится массив правильных чисел, которые были введены.

**Заключение:**

В ходе выполнения лабораторной работы я научился создавать и обрабатывать собственные исключения в Java, что позволило мне лучше понять механизмы обработки ошибок. Я разработал программу, которая запрашивает у пользователя ввод 10 чисел, проверяет их на простоту, и обрабатывает ввод некорректных данных с помощью исключений. Эта работа показала мне, как важна валидация пользовательского ввода, а также значимость предоставления обратной связи пользователю в случае ошибок. Полученные знания и навыки будут полезны при разработке более сложных приложений, где требуется эффективное управление исключениями и обработка ошибок.