**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Ордена трудового Красного Знамени федеральное государственное бюджетное

образовательное учреждение высшего образования

**«Московский технический университет связи и информатики»**

Факультет «Информационные технологии»

Кафедра «Математическая кибернетика и информационные технологии»

Дисциплина: «Информационные технологии и программирование»

**Лабораторная работа №4**

**«Обработка исключений»**

Выполнил:

студент группы БВТ2402

Аралушкин Максим Дмитриевич

Руководитель:

Мосева Марина Сергеевна

Москва

2025

# Цель работы

Освоение работы с обработками исключений на Java. Изучение и создание обработчиков исключений.

# Индивидуальное задание

1. Написать программу, которая будет находить среднее арифметическое элементов массива. При этом программа должна обрабатывать ошибки, связанные с выходом за границы массива и неверными данными (например, если элемент массива не является числом).
2. Написать программу, которая будет копировать содержимое одного файла в другой. При этом программа должна обрабатывать возможные ошибки, связанные с открытием и закрытием файлов (вариант 1)
3. Создайте класс CustomDivisionException, который будет использоваться для обработки исключений при делении на ноль. Напишите программу, которая делит два числа, и, если происходит деление на ноль, выбрасывайте исключение CustomDivisionException. Создайте обработчик исключений, который логирует информацию о каждом выброшенном исключении в текстовый файл.

# Ход работы

Задание №1:

package lab4;

public class ArrayAverage {

    public static void main(String[] args) {

        Object[] arr = {1, 2, 3, 4, 5, "9"};

        int summ = 0;

        try {

            for (int i = 0; i < arr.length; i++) {

                if (arr[i] instanceof String) {

                    throw new ClassCastException("Ошибка: элемент массива не является числом.");

                }

                summ += (int) arr[i];

            }

            System.out.println("Среднее арифм массива: " + summ/arr.length);

        } catch (IndexOutOfBoundsException e) {

            System.out.println("Ошибка: индекс вне массива.");

        } catch (ClassCastException e) {

            System.out.println(e.getMessage());

        } catch (Exception e) {

            System.out.println("Другой вид ошибки.");

        }

    }

}

*// Массив из строк идёт по перебору int - выдать ошибку*

Задание №2:

package lab4;

import java.nio.file.\*;

import java.io.\*;

public class Copying {

    public static void main(String[] args) {

        Path file = Paths.get("Files.txt");

        Path copy = Paths.get("CopyFile.txt");

        try {

            Files.copy(file, copy, StandardCopyOption.REPLACE\_EXISTING);

            System.out.println("Файл File.txt скопирован в CopyFile.txt");

        } catch (NoSuchFileException e) {

            System.out.println("Ошибка: файл " + e.getMessage() + " не найден.");

        } catch (IOException e) {

            System.out.println("Ошибка: " + e.toString());

        }

    }

}

*// Не сущ файл - выдать ошибку*

Задание №3:

package lab4;

import java.nio.file.\*;

import java.io.\*;

class CustomDivisionException extends Exception{

    public CustomDivisionException(String message){

        super(message);

    }

}

public class DivisionOnZero {

    public static double divide(double a, double b) throws CustomDivisionException {

        if (b==0) {

            throw new CustomDivisionException("Ошибка: деление на ноль.");

        }

        return a/b;

    }

public static void logsError(String message) {

    Path logs = Paths.get("logs.txt");

    try {

        Files.write(logs, message.getBytes(), StandardOpenOption.CREATE, StandardOpenOption.APPEND);

    } catch (IOException e) {

        System.out.println("Ошибка не записана в логи: " + e.getMessage());

    }

}

    public static void main(String[] args) {

        try {

            double result = divide(10, 0);

            System.out.println("Результат деления: " + result);

        } catch (CustomDivisionException e) {

            System.out.println(e.getMessage());

            logsError(e.getMessage() + "\n");

        } catch (Exception e) {

            System.out.println("Ошибка: " + e.getMessage());

            logsError(e.getMessage());

        }

    }

}

# Вывод

В ходе выполнения данной лабораторной работы я освоил работу с обработками исключений на Java. Изучил и создал обработчик исключений. Выполнил индивидуальные задание, с использованием исключений.