

Міністерство освіти і науки України  
Національний університет «Львівська політехніка»  
Кафедра «Електронних обчислювальних машин»



Звіт  
з лабораторної роботи № 5  
з дисципліни: «Кросплатформенні засоби програмування»  
на тему: «ФАЙЛИ У JAVA»

**Виконав:**

студент групи КІ-306

*Хмільовський С. Р.*

**Прийняв:**

доцент кафедри ЕОМ

*Іванов Ю. С.*

**Мета роботи:** оволодіти навиками використання засобів мови Java для роботи з потоками і файлами.

### **Завдання (варіант № 22)**

1. Створити клас, що реалізує методи читання/запису у текстовому і двійковому форматах результатів роботи класу, що розроблений у лабораторній роботі No4. Написати програму для тестування коректності роботи розробленого класу.
2. Для розробленої програми згенерувати документацію.
3. Завантажити код на GitHub згідно методичних вказівок по роботі з GitHub.
4. Скласти звіт про виконану роботу з приведенням тексту програми, результату її виконання та фрагменту згенерованої документації та завантажити його у ВНС.
5. Дати відповідь на контрольні запитання.

### **Вихідний код програми:**

#### *Файл FioApp.java*

```
//package KI306.Khmilovskiy.Lab5;

import java.io.*;
import java.util.*;

/**
 * FioApp used for calculation and store results in both text and binary
 * formats.
 * @author Khmilovskiy Stanislaw
 * @version 1.0
 * @since 1.0
 */
public class FioApp {
    public static void main(String[] args) throws FileNotFoundException,
    IOException
    {
        CalcWFio obj = new CalcWFio();
        Scanner s = new Scanner(System.in);

        System.out.print("Введіть значення \"x\" [double]: ");
        double x = s.nextDouble();

        obj.calculate(x);
        System.out.println("Результат обчислення виразу: " + obj.getResult());

        // Запис результату у текстовий файл
        obj.writeResTxt("TextResult.txt");
        System.out.println("\nРезультат був записаний у текстовий файл!");
    }
}
```

```

        // Запис результату у бінарний файл
        obj.writeResBin("BinaryResult.bin");
        System.out.println("Результат був записаний у бінарний файл!");

        // Читання результату з бінарного файлу
        String binFileName = "BinaryResult.bin";
        obj.readResBin("BinaryResult.bin");
        System.out.println("\nРезультат був зчитаний з бінарного файлу: " +
binFileName);
        System.out.println("Зчитаний результат з бінарного файлу: " +
obj.getResult());

        // Читання результату з текстового файлу
        String txtFileName = "TextResult.txt";
        obj.readResTxt("TextResult.txt");
        System.out.println("\nРезультат був зчитаний з текстового файлу: " +
txtFileName);
        System.out.println("Зчитаний результат з текстового файлу: " +
obj.getResult());
    }
}

/**
 * A class for performing calculations and handling file I/O operations.
 */
class CalcWFio
{
    /**
     * Writes the result to a text file.
     *
     * @param fName The name of the text file to write to.
     * @throws FileNotFoundException If the file cannot be found.
     */
    public void writeResTxt(String fName) throws FileNotFoundException
    {
        PrintWriter f = new PrintWriter(fName);
        f.printf("%f ", result);
        f.close();
    }

    /**
     * Reads the result from a text file.
     *
     * @param fName The name of the text file to read from.
     */
    public void readResTxt(String fName)
    {
        try
        {
            File f = new File (fName);
            if (f.exists())
            {
                Scanner s = new Scanner(f);
                result = s.nextDouble();
                s.close();
            }
            else
                throw new FileNotFoundException("File " + fName + "not found");
        }
        catch (FileNotFoundException ex)
        {
            System.out.print(ex.getMessage());
        }
    }
}

```

```

/**
 * Writes the result to a binary file.
 *
 * @param fName The name of the binary file to write to.
 * @throws FileNotFoundException If the file cannot be found.
 * @throws IOException          If an I/O error occurs.
 */
public void writeResBin(String fName) throws FileNotFoundException,
IOException
{
    DataOutputStream f = new DataOutputStream(new FileOutputStream(fName));
    f.writeDouble(result);
    f.close();
}

/**
 * Reads the result from a binary file.
 *
 * @param fName The name of the binary file to read from.
 * @throws FileNotFoundException If the file cannot be found.
 * @throws IOException          If an I/O error occurs.
 */
public void readResBin(String fName) throws FileNotFoundException,
IOException
{
    DataInputStream f = new DataInputStream(new FileInputStream(fName));
    result = f.readDouble();
    f.close();
}

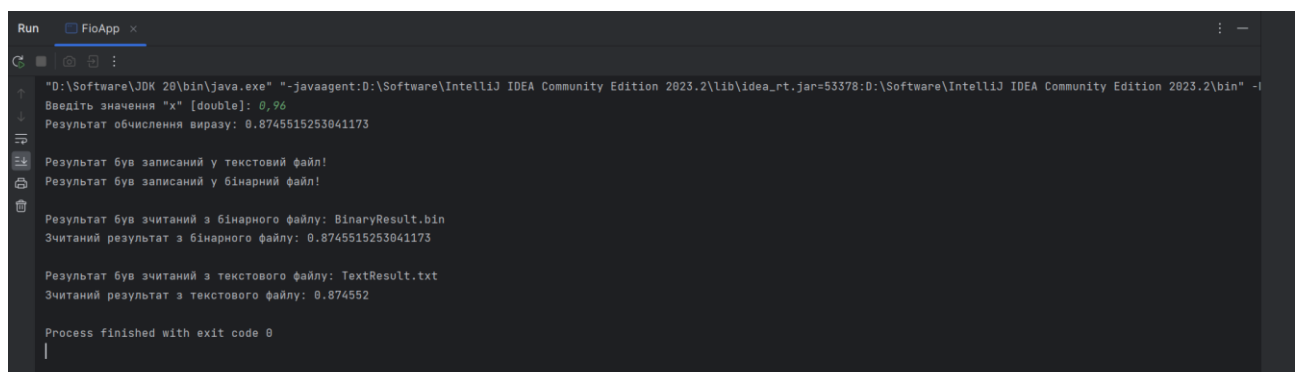
/**
 * Calculates a result based on the given input.
 *
 * @param x The input value for the calculation.
 */
public void calculate(double x)
{
    result = EquationSolver.calculateExpression(x);
}

/**
 * Gets the calculated result.
 *
 * @return The calculated result.
 */
public double getResult()
{
    return result;
}

private double result;
}

```

## Результат виконання програми:



```

Run FioApp x
D:\Software\JDK 20\bin\java.exe "-javaagent:D:\Software\IntelliJ IDEA Community Edition 2023.2\lib\idea_rt.jar=53378:D:\Software\IntelliJ IDEA Community Edition 2023.2\bin" -I
Введіть значення "x" [double]: 0,96
Результат обчислення виразу: 0.8745515253041173

Результат був записаний у текстовий файл!
Результат був записаний у бінарний файл!

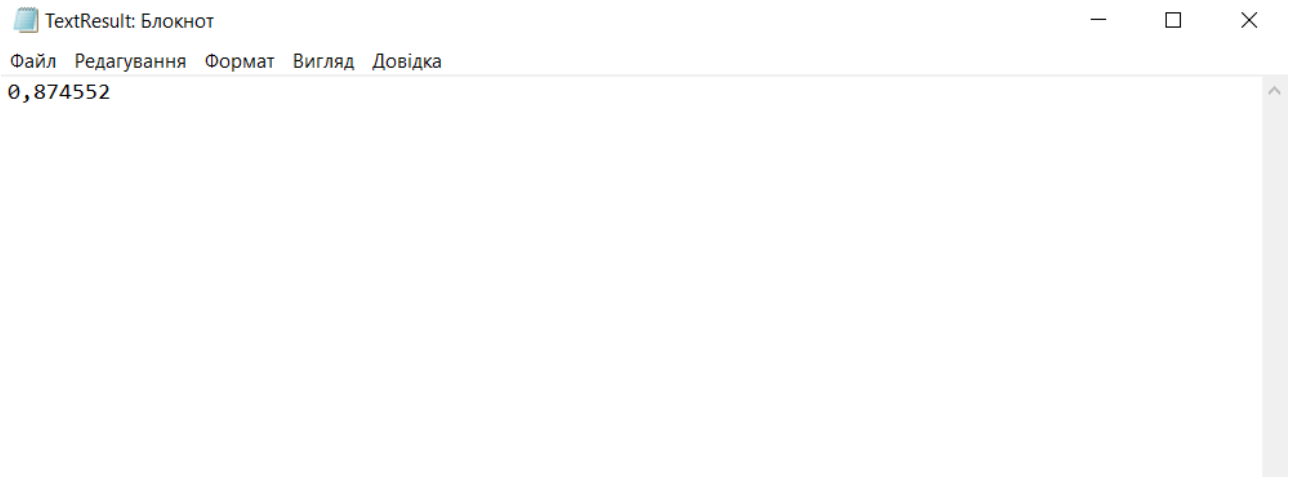
Результат був зчитаний з бінарного файлу: BinaryResult.bin
Зчитаний результат з бінарного файлу: 0.8745515253041173

Результат був зчитаний з текстового файлу: TextResult.txt
Зчитаний результат з текстового файлу: 0.874552

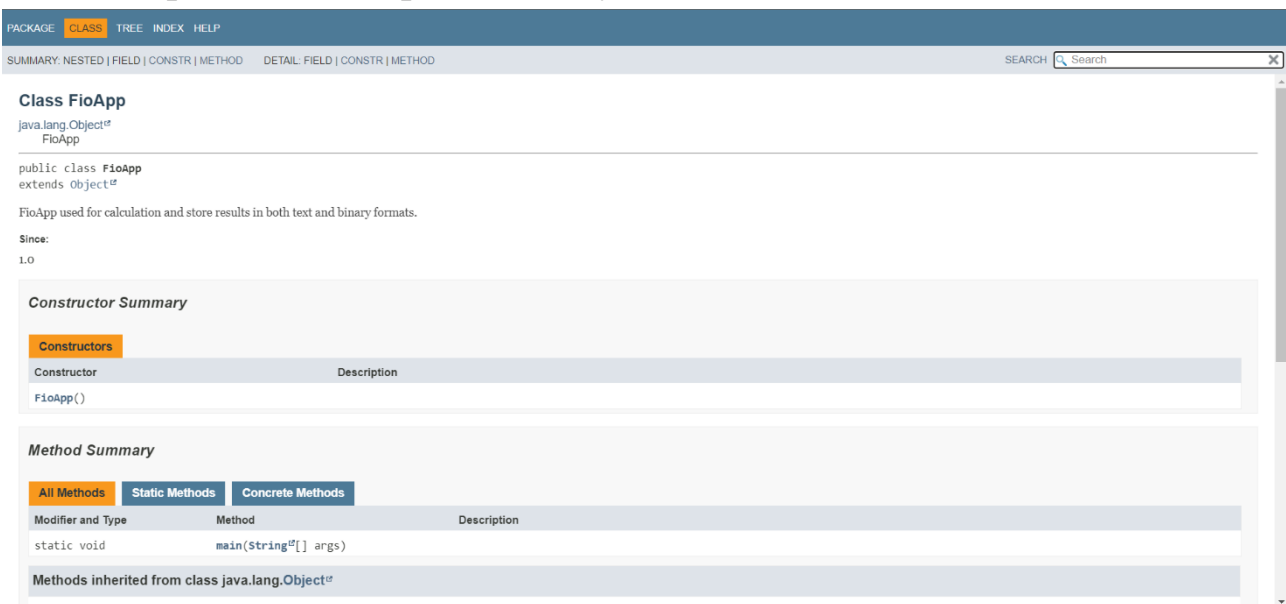
Process finished with exit code 0

```

## Вміст файлу TextResult.txt:



## Фрагмент згенерованої документації JavaDoc:



## Відповіді на контрольні запитання:

1. Розкрийте принципи роботи з файловою системою засобами мови Java.

Java надає набір класів та інтерфейсів для роботи з файлами та файловою системою. Основні операції включають в себе читання та запис даних до файлу, перевірку існування файлів, видалення, переміщення та інші операції. Для роботи з файлами в Java використовують класи "File", "FileInputStream", "FileOutputStream", "BufferedReader", "BufferedWriter", "RandomAccessFile" та інші.

2. Охарактеризуйте клас Scanner.

“Scanner” - це клас у пакеті java.util, який дозволяє зчитувати різні типи даних з різних джерел, таких як клавіатура або файли. Він надає методи для парсингу і зчитування даних, включаючи цілі числа, числа з плаваючою точкою, рядки та інші типи.

### 3. Наведіть приклад використання класу Scanner.

```
import java.util.Scanner;

public class ScannerExample {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter an integer: ");

        int number = scanner.nextInt();

        System.out.println("You entered: " + number);

        scanner.close();

    }

}
```

### 4. За допомогою якого класу можна здійснити запис у текстовий потік?

Для запису у текстовий потік в Java використовують клас “PrintWriter”. Він дозволяє записувати текстові дані до файлу або іншого виводу.

### 5. Охарактеризуйте клас PrintWriter.

“PrintWriter” - це клас, який надає методи для запису рядків та інших типів даних до текстового потоку, такого як файл або консоль. Він автоматично виконує буферизацію та перетворення даних у рядки.

### 6. Розкрийте методи читання/запису двійкових даних засобами мови Java.

Для читання та запису двійкових даних в Java використовують класи “DataInputStream” та “DataOutputStream”. Вони дозволяють прочитати та записати дані в двійковому форматі, такі як цілі числа, числа з плаваючою точкою, символи та інші типи.

### 7. Призначення класів DataInputStream і DataOutputStream.

“DataInputStream” та “DataOutputStream” - це класи, які надають методи для читання та запису даних в двійковому форматі. Вони дозволяють зберігати дані в бінарних файлах та передавати їх мережею.

8. Який клас мови Java використовується для здійснення довільного доступу до файлів.

Для здійснення довільного доступу до файлів в Java використовують клас “RandomAccessFile”. Він дозволяє читати та записувати дані у файлі за допомогою вказівника на поточне положення у файлі.

9. Охарактеризуйте клас RandomAccessFile.

“RandomAccessFile” - це клас, який надає можливість читати та записувати дані в файлі за допомогою довільного доступу, включаючи переміщення вказівника в файлі.

10. Який зв'язок між інтерфейсом DataOutput і класом DataOutputStream?

“DataOutput” - це інтерфейс, який визначає методи для запису даних в двійковому форматі. Клас “DataOutputStream” імплементує цей інтерфейс і надає реалізацію методів для запису різних типів даних в двійковому форматі до потоку даних.

### **Висновок:**

На даній лабораторній роботі я оволодів навичками використання засобів мови Java для роботи з потоками і файлами.