Міністерство освіти і науки України Національний університет «Львівська політехніка»

Кафедра «ЕОМ»



Звіт

до лабораторної роботи $N \hspace{-.08cm} \underline{\hspace{0.08cm}} \hspace{0.1cm} 5$

з дисципліни: «Кросплатформні засоби програмування»

На тему: «Файли у java»

Виконав:

студент групи KI-307 Возний А. О.

Перевірив:

доцент кафедри ЕОМ Іванов Ю. С. **Мета роботи:** оволодіти навиками використання засобів мови Java для роботи з потоками і файлами.

Завдання:

- 1. Створити клас, що реалізує методи читання/запису у текстовому і двійковому форматах результатів роботи класу, що розроблений у лабораторній роботі №4. 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.

Варіант завдання: y = cos(x)/sin(x)

Лістинг програми:

Клас Equations Lab5

```
package ki307.voznyi.lab5;

import java.io.*;

/**

* <u>Kлас</u> Equations_Lab5 <u>для обчислення та запису результату функції cos(x)</u>/sin(x) у текстовий та <u>бінарний файл.</u>

*/

public class Equations_Lab5 {
    private double x;
    private double x_rad;
    private PrintWriter fout;

/**

* <u>KOHCTPYKTOP 3a 3aMOBYYBAHHЯM, iHiqiani3ye значення x та x_rad.</u>

*/

public Equations_Lab5() {
    this.x_rad = 0;
    }

/**

* <u>KOHCTPYKTOP 3 параметром, iHiqiani3ye значення x та x_rad зазначеним значенням</u>

x y <u>градусах.</u>

* <u>Popanycax.</u>

*

* <u>@param x Значення x y градусах.</u>

*/

public Equations_Lab5(double x) {
    this.x_ = x;
    this.x_rad = x * 3.14159 / 180;
}

/**

* <u>Metod для обчислення виразу cos(x)</u>/sin(x).
```

```
* @return Результат обчислення виразу cos(x)/sin(x).
     * @throws ArithmeticException Виняток, якщо sin(x) дорівнює 0.
    public double Calc() throws ArithmeticException {
       if (Math.sin(x_rad) == 0) {
            throw new ArithmeticException("Exeption: sin(x) is equal to 0!\n");
           return Math.cos(x_rad) / Math.sin(x_rad);
    }
    * Метод для запису результату обчислення у текстовий та бінарний файл.
    * @param out file
    * @param out_file_bin <u>Шлях до</u> бінарного файлу.
    public void in_files(String out_file, String out_file_bin) {
            fout = new PrintWriter(new BufferedWriter(new FileWriter(out_file)));
            fout.print("cos(x)/sin(x) = " + Calc() + "\n");
            fout.close();
            DataOutputStream fout_bin = new DataOutputStream(new
FileOutputStream(out_file_bin));
            fout_bin.writeChars("cos(x)/sin(x) .bin file = " + Calc() + "\n");
            fout bin.close();
        } catch (IOException e) {
            System.err.println("Can't use the file!\n");
        } catch (ArithmeticException ae) {
           System.out.print(ae.getMessage());
    * Метод для закриття текстового файлу після використання.
    public void close file() {
       fout.close();
```

Клас FileIO Class

```
package ki307.voznyi.lab5;
import java.io.*;

/**
    * Kлас FileIO_Class для введення та виведення значень до текстового та бінарного файлів.
    */
public class FileIO_Class {
    private File file;
    private File file_bin;
    /**
          * Конструктор класу, ініціалізує шляхи до текстового та бінарного файлів.
```

```
* # @param file
     * @param file_bin Шлях до бінарного файлу.
    FileIO Class(File file, File file bin) {
       this.file = file;
        this.file_bin = file_bin;
    }
      Метод для введення значення у текстовий та бінарний файл.
     * @param x <u>Значення для введення</u>.
    public void inFile(double x) {
        PrintWriter fin = null;
        DataOutputStream fin_bin = null;
            fin = new PrintWriter(new BufferedWriter(new FileWriter(file)));
            fin_bin = new DataOutputStream(new FileOutputStream(file bin));
            fin.println(x);
            fin.close();
            fin_bin.writeDouble(x);
            fin_bin.close();
        } catch (IOException e) {
            System.err.println("Can't open the file!" + e);
      Метод для виведення значень з текстового та бінарного файлів.
   public void outFile() {
        BufferedReader fout = null;
        DataInputStream fout_bin = null;
            fout = new BufferedReader(new FileReader(file));
            fout_bin = new DataInputStream(new FileInputStream(file_bin));
            System.out.println("Result of reading file: \n" + fout.readLine() +
"\n");
            fout.close();
            System.out.println("Result of reading .bin file: \n" +
fout_bin.readDouble() + "\n");
            fout_bin.close();
        } catch (IOException e) {
            System.err.println("Error:" + e);
    }
```

Клас FioJava

```
package ki307.voznyi.lab5;
import java.io.File;
import java.io.IOException;
import java.util.Scanner;
```

```
public class FioJava {
    * Основний метод програми.
    * @param <u>args</u> <u>Аргументи</u> <u>командного</u> <u>рядка</u>.
    * @throws IOException Виняток, який може виникнути при роботі з файлами.
   public static void main(String[] args) throws IOException {
       double res;
       File file = new File("Lab5 Voznyi.txt");
       File file_bin = new File("Lab5_Voznyi_binary.bin");
       Scanner input = new Scanner(System.in);
       System.out.print("Input X: ");
       x = input.nextDouble();
       Equations_Lab5 eq = new Equations_Lab5(x);
       res = eq.Calc();
       System.out.print("cos(x)/sin(x) = " + res + "\n\n");
       FileIO_Class fio = new FileIO_Class(file, file_bin);
       fio.inFile(res);
       fio.outFile();
       input.close();
```

Результат виконання програми:

```
Input X: 50
cos(x)/sin(x) = 0.8391008872755654

Result of reading file:
0.8391008872755654

Result of reading .bin file:
0.8391008872755654
```

Рис. 1. Результат виконання програми

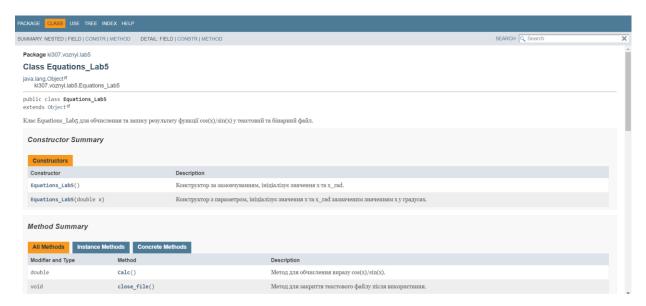


Рис. 2. Згенерована документація до класу Клас Equations Lab5

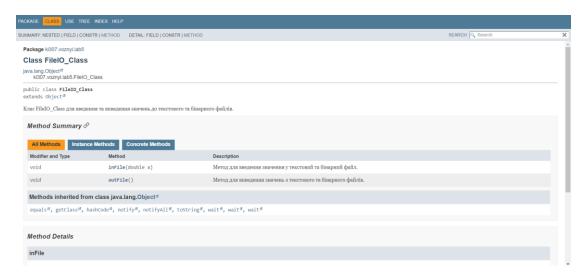


Рис. 3. Згенерована документація до класу FileIO_Class

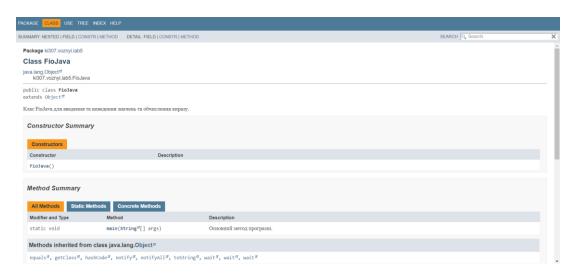


Рис. 4. Згенерована документація до класу FioJava

Відповіді на контрольні питання:

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

Принципи роботи з файловою системою засобами мови Java базуються на використанні класів для роботи з файлами, таких як 'File', 'FileInputStream', 'FileOutputStream', 'BufferedReader', 'BufferedWriter', 'Scanner', 'PrintWriter', 'DataInputStream', 'DataOutputStream', 'RandomAccessFile'. З цими класами ви можете створювати, читати та записувати файли, працювати з текстовими і бінарними даними, а також здійснювати оптимізований доступ до файлів.

2. Охарактеризуйте клас Scanner.

Клас `Scanner` використовується для зчитування даних з різних джерел, таких як клавіатура, файли, рядки і т. д. Він дозволяє вказати роздільний символ і перетворити зчитані дані у відповідні типи даних.

3. Наведіть приклад використання класу Scanner.

```
Scanner scanner = new Scanner(System.in);
System.out.print("Введіть число: ");
int number = scanner.nextInt();
System.out.println("Ви ввели: " + number);
```

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

'PrintWriter'

5. Охарактеризуйте клас PrintWriter.

Клас 'PrintWriter' використовується для запису даних у текстовий файл. Він дозволяє друкувати рядки та дані у файл.

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

`DataInputStream` i `DataOutputStream`. Вони дозволяють зчитувати і записувати різні типи даних (наприклад, цілі числа, дійсні числа) у бінарному форматі.

7. Призначення класів DataInputStream i DataOutputStream.

Класи `DataInputStream` i `DataOutputStream` призначені для читання та запису бінарних даних. Вони надають методи для зчитування і запису різних типів даних, таких як `int`, `double`, `boolean` і т. д., у бінарному форматі.

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

Для довільного доступу до файлів використовується клас 'RandomAccessFile'. Він дозволяє читати та записувати дані у файлі з будь-якої позиції, а не обов'язково починаючи з початку.

9. Охарактеризуйте клас RandomAccessFile.

Клас `RandomAccessFile` використовується для довільного доступу до файлів. Він дозволяє читати і записувати дані з будь-якої позиції в файлі.

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

Інтерфейс `DataOutput` визначає методи для запису даних у бінарний потік. Клас `DataOutputStream` реалізує цей інтерфейс і дозволяє записувати дані у бінарний формат.

Висновок: під час виконання цієї лабораторної роботи я оволодів навиками використання засобів мови Java для роботи з потоками і файлами.