

Міністерство освіти і науки України
Національний університет «Львівська політехніка»

Кафедра «ЕОМ»



Звіт

до лабораторної роботи № 5

з дисципліни: «Кросплатформні засоби програмування»

На тему: «Файли у java»

Виконав:

студент групи КІ-307

Возний А. О.

Перевірив:

доцент кафедри ЕОМ

Іванов Ю. С.

Львів – 2023

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.*;

/**
 * Клас Equations_Lab5 для обчислення та запису результату функції  $\cos(x)/\sin(x)$  у
 * текстовий та бінарний файл.
 */
public class Equations_Lab5 {
    private double x;
    private double x_rad;
    private PrintWriter fout;

    /**
     * Конструктор за замовчуванням, ініціалізує значення x та x_rad.
     */
    public Equations_Lab5() {
        this.x = 0;
        this.x_rad = 0;
    }

    /**
     * Конструктор з параметром, ініціалізує значення x та x_rad зазначеним значенням
     * x у градусах.
     *
     * @param x Значення x у градусах.
     */
    public Equations_Lab5(double x) {
        this.x = x;
        this.x_rad = x * 3.14159 / 180;
    }

    /**
     * Метод для обчислення виразу  $\cos(x)/\sin(x)$ .
     */
}
```

```

*
* @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("Exception: sin(x) is equal to 0!\n");
    } else {
        return Math.cos(x_rad) / Math.sin(x_rad);
    }
}

/**
 * Метод для запису результату обчислення у текстовий та бінарний файл.
 */
* @param out_file Шлях до текстового файлу.
* @param out_file_bin Шлях до бінарного файлу.
*/
public void in_files(String out_file, String out_file_bin) {
    try {
        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.*;

/**
 * Клас 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    Шлях до текстового файлу.
* @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 Значення для введення.
 */
public void inFile(double x) {
    PrintWriter fin = null;
    DataOutputStream fin_bin = null;
    try {
        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;
    try {
        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;

```

```

/**
 * Клас FioJava для введення та виведення значень та обчислення виразу.
 */
public class FioJava {

    /**
     * Основний метод програми.
     *
     * @param args Аргументи командного рядка.
     * @throws IOException Виняток, який може виникнути при роботі з файлами.
     */
    public static void main(String[] args) throws IOException {
        double x;
        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

PACKAGE

CLASS

USE

TREE

INDEX

HELP

SUMMARY

NESTED

FIELD

CONSTR

METHOD

DETAIL

FIELD

CONSTR

METHOD

SEARCH

Package

ki307.voznyi.lab5

Class

Equations_Lab5

java.lang.Object

ki307.voznyi.lab5.Equations_Lab5

```

public class Equations_Lab5
extends Object

```

Клас Equations_Lab5 для обчислення та запису результату функції $\cos(x)/\sin(x)$ у текстовий та бінарний файл.

Constructor Summary

Constructors

Constructor	Description
<code>Equations_Lab5()</code>	Конструктор за замовчуванням, ініціалізує значення x та x_rad .
<code>Equations_Lab5(double x)</code>	Конструктор з параметром, ініціалізує значення x та x_rad зазначеним значенням x у градусах.

Method Summary

All Methods

Instance Methods

Concrete Methods

Modifier and Type	Method	Description
double	<code>calc()</code>	Метод для обчислення виразу $\cos(x)/\sin(x)$.
void	<code>close_file()</code>	Метод для закриття текстового файлу після використання.

Рис. 2. Згенерована документація до класу Клас Equations_Lab5

PACKAGE

CLASS

USE

TREE

INDEX

HELP

SUMMARY

NESTED

FIELD

CONSTR

METHOD

DETAIL

FIELD

CONSTR

METHOD

SEARCH

Package

ki307.voznyi.lab5

Class

FileIO_Class

java.lang.Object

ki307.voznyi.lab5.FileIO_Class

```

public class FileIO_Class
extends Object

```

Клас FileIO_Class для введення та виведення значень до текстового та бінарного файлів.

Method Summary

All Methods

Instance Methods

Concrete Methods

Modifier and Type	Method	Description
void	<code>inFile(double x)</code>	Метод для введення значення у текстовий та бінарний файл.
void	<code>outFile()</code>	Метод для виведення значень з текстового та бінарного файлів.

Methods inherited from class java.lang.Object

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

Method Details

inFile

Рис. 3. Згенерована документація до класу FileIO_Class

PACKAGE

CLASS

USE

TREE

INDEX

HELP

SUMMARY

NESTED

FIELD

CONSTR

METHOD

DETAIL

FIELD

CONSTR

METHOD

SEARCH

Package

ki307.voznyi.lab5

Class

FioJava

java.lang.Object

ki307.voznyi.lab5.FioJava

```

public class FioJava
extends Object

```

Клас FioJava для введення та виведення значень та обчислення виразу.

Constructor Summary

Constructors

Constructor	Description
<code>FioJava()</code>	

Method Summary

All Methods

Static Methods

Concrete Methods

Modifier and Type	Method	Description
static void	<code>main(String[] args)</code>	Основний метод програми.

Methods inherited from class java.lang.Object

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

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

7. Призначення класів DataInputStream і DataOutputStream.

Класи `'DataInputStream'` і `'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.