

Міністерство освіти і науки України

Національний університет “Львівська політехніка”

Кафедра ЕОМ



Звіт

З лабораторної роботи №5

Варіант – 9

З дисципліни: «Кросплатформні засоби програмування»

На тему: «ФАЙЛИ У JAVA »

Виконав: ст. гр. КІ-305

Заставний Р.А.

Прийняв:

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:

9. $y = \text{tg}(x)/3x$

Код програми:

```
package K1305.Zastavnyi.Lab5;

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

/**
 * Клас EquationsApp демонструє обчислення та операції з файлами.
 * Використовує клас CalcWFio для обчислень та обробки винятків.
 * Це консольний додаток, де користувач вводить дані, а результат виводиться та зберігається в файлах.
 */
public class EquationsApp {

    /**
     * Головний метод класу EquationsApp.
     * Створює екземпляр CalcWFio, отримує введення користувача, виконує обчислення та демонструє операції з файлами.
     */
    *
    * @param args Масив аргументів командного рядка.
    */
    public static void main(String[] args) {
        CalcWFio obj = new CalcWFio();
        Scanner s = new Scanner(System.in);

        try {
            System.out.print("Введіть дані: ");
            double data = s.nextDouble();

            obj.calculate(data);
            System.out.println("Результат: " + obj.getResult());

            obj.writeResTxt("textRes.txt");
            obj.writeResBin("BinRes.bin");

            obj.readResBin("BinRes.bin");
            System.out.println("Результат: " + obj.getResult());

            obj.readResTxt("textRes.txt");
            System.out.println("Результат: " + obj.getResult());
        } catch (FileNotFoundException ex) {
            System.out.println("Причина винятку: Файл не знайдено");
        } catch (IOException ex) {
            System.out.println("Причина винятку: Помилка введення/виведення");
        }
    }
}
```

```

    } catch (ArithmeticException ex) {
        System.out.println("Причина винятку: " + ex.getMessage());
    }
}

/**
 * Клас CalcWFio виконує обчислення та операції з файлами для результатів.
 */
class CalcWFio {
    private double result;

    /**
     * Записує результат в текстовий файл.
     *
     * @param fName Ім'я файлу для запису результату.
     * @throws FileNotFoundException Виникає, якщо файл не знайдено.
     */
    public void writeResTxt(String fName) throws FileNotFoundException {
        PrintWriter f = new PrintWriter(fName);
        f.printf("%f ", result);
        f.close();
    }

    /**
     * Зчитує результат з текстового файлу.
     *
     * @param fName Ім'я файлу для зчитування результату.
     */
    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("Файл " + fName + " не знайдено");
            }
        } catch (FileNotFoundException ex) {
            System.out.println(ex.getMessage());
        }
    }

    /**
     * Записує результат в двійковий файл.
     *
     * @param fName Ім'я файлу для запису результату.
     * @throws FileNotFoundException Виникає, якщо файл не знайдено.
     * @throws IOException Виникає, якщо виникає помилка введення/виведення.
     */
    public void writeResBin(String fName) throws FileNotFoundException, IOException {
        DataOutputStream f = new DataOutputStream(new FileOutputStream(fName));
        f.writeDouble(result);
        f.close();
    }

    /**
     * Зчитує результат з двійкового файлу.
     *
     * @param fName Ім'я файлу для зчитування результату.
     * @throws FileNotFoundException Виникає, якщо файл не знайдено.
     * @throws IOException Виникає, якщо виникає помилка введення/виведення.
     */
}

```

```

public void readResBin(String fName) throws FileNotFoundException, IOException {
    DataInputStream f = new DataInputStream(new FileInputStream(fName));
    result = f.readDouble();
    f.close();
}

/**
 * Виконує обчислення на основі введених даних.
 *
 * @param x Вхідні дані для обчислення виразу.
 */
public void calculate(double x) {
    double y, rad;
    rad = x * Math.PI / 180.0;

    if (rad == Math.PI / 2.0 || rad == -Math.PI / 2.0) {
        throw new ArithmeticException("Причина винятку: Неправильне значення даних для обчислення тангенсу");
    } else if (x == 0) {
        throw new ArithmeticException("Причина винятку: Дані = 0");
    } else {
        y = (Math.tan(rad)) / 3 * x;
        result = y;
    }
}

/**
 * Повертає результат обчислення.
 *
 * @return Результат обчислення.
 */
public double getResult() {
    return result;
}
}

```

Результата роботи програми:

```

"C:\Program Files\Microsoft\jdk-17.0.8.101-hotspot\bin\java.exe" "-javaagent
Enter data: 10
Result is: 0.5877566023615499
Result is: 0.5877566023615499
Result is: 0.587757

Process finished with exit code 0
|

```

Фрагмент згенерованої документації

Select Administrator: Командный рядок

```
C:\Users\fynti\Desktop\КЭП\Lab5ZastavnyiKI305>cd out
C:\Users\fynti\Desktop\КЭП\Lab5ZastavnyiKI305\out>cd ..
C:\Users\fynti\Desktop\КЭП\Lab5ZastavnyiKI305>cd src
C:\Users\fynti\Desktop\КЭП\Lab5ZastavnyiKI305\src>cd KI305
C:\Users\fynti\Desktop\КЭП\Lab5ZastavnyiKI305\src\KI305>cd Zastavnyi
C:\Users\fynti\Desktop\КЭП\Lab5ZastavnyiKI305\src\KI305\Zastavnyi>cd Lab5
C:\Users\fynti\Desktop\КЭП\Lab5ZastavnyiKI305\src\KI305\Zastavnyi\Lab5>javadoc EquationsApp.java
Loading source file EquationsApp.java...
Constructing Javadoc information...
Building index for all the packages and classes...
Standard Doclet version 17.0.8.1+1-LTS
Building tree for all the packages and classes...
Generating .\KI305\Zastavnyi\Lab5\EquationsApp.html...
Generating .\KI305\Zastavnyi\Lab5\package-summary.html...
Generating .\KI305\Zastavnyi\Lab5\package-tree.html...
Generating .\overview-tree.html...
Building index for all classes...
Generating .\allclasses-index.html...
Generating .\allpackages-index.html...
Generating .\index-all.html...
Generating .\index.html...
Generating .\help-doc.html...
C:\Users\fynti\Desktop\КЭП\Lab5ZastavnyiKI305\src\KI305\Zastavnyi\Lab5>
```

Package KI305.Zastavnyi.Lab5

Class EquationsApp

[java.lang.Object](#)

KI305.Zastavnyi.Lab5.EquationsApp

```
public class EquationsApp
```

```
extends Object
```

Клас EquationsApp демонструє обчислення та операції з файлами. Використовує клас CalcWfio для обчислень та обробки винятків. Це консольний додаток, де користувач вводить дані, а результат виводиться та зберігається в файлах.

Constructor Summary

Constructors

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

Method Summary

All Methods

Static Methods

Concrete Methods

Modifier and Type	Method	Description
static void	<code>main(String¹²[] args)</code>	Головний метод класу EquationsApp.

Methods inherited from class `java.lang.Object`

```
clone2, equals2, finalize2, getClass2, hashCode2, notify2, notifyAll2, toString2, wait2, wait2, wait2
```

Constructor Details

EquationsApp

```
public EquationsApp()
```

Висновок: на цій лабораторній роботі я олодів навиками використання засобів мови Java для роботи з потоками і файлами.