Міністерство освіти і науки України

Національний університет “Львівська політехніка”

Кафедра ЕОМ



ЗВІТ

до лабораторної роботи №4

З дисципліни: «Кросплатформні засоби програмування»

На тему: «ВИКЛЮЧЕННЯ»

Варіант 20

**Виконав:**

ст. групи КІ-306

Петрук О.С.

**Прийняв:**

доцент кафедри ЕОМ

Олексів М.В.

Львів – 2024

**Мета:** оволодіти навиками використання механізму виключень при написанні программовою Java.

**Завдання:**

1. Створити клас, що реалізує метод обчислення виразу заданого варіантом. Написати на мові Java та налагодити програму-драйвер для розробленого класу. Результат обчислень записати у файл. При написанні програми застосувати механізм виключень для виправлення помилкових ситуацій, що можуть виникнути в процесі виконання програми. Програма має розміщуватися в пакеті Група.Прізвище.Lab4 та володіти коментарями, які дозволять автоматично згенерувати документацію до розробленого пакету.

2. Автоматично згенерувати документацію до розробленого пакету.

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.

**Функція згідно варіанту №20 – « y = tan(x)\*cot(2x) »**

**GitHub Repository:**

**Хід роботи**

Код програми:

***ExpressionCalculator.java***

package KI\_306.Petruk.Lab4;

import java.io.FileWriter;

import java.io.IOException;

import java.util.InputMismatchException;

/\*\*

\* Class for calculating the expression y=tg(x)ctg(2x).

\* Demonstrates exception handling and writing results to a file.

\*/

public class ExpressionCalculator {

/\*\*

\* Calculates the expression y=tg(x)ctg(2x).

\*

\* @param x the value of the variable x

\* @return the result of the calculation

\* @throws IllegalArgumentException if x = 0, as the expression is undefined

\*/

public double calculate(double x) throws IllegalArgumentException {

if (x == 0) {

throw new IllegalArgumentException("The value of x cannot be 0, as the expression is undefined.");

}

return Math.tan(x) \* (1/Math.tan(2\*x));

}

/\*\*

\* Validates if the given input is a number.

\*

\* @param input the input value as a string

\* @return the numeric value of x if it is a valid number

\* @throws InputMismatchException if the input is not a number

\*/

public double validateAndParse(String input) throws InputMismatchException {

try {

return Double.parseDouble(input);

} catch (NumberFormatException e) {

throw new InputMismatchException("The input value must be a number.");

}

}

/\*\*

\* Writes the calculation result to a file.

\*

\* @param result the result of the calculation

\* @param filePath the path to the file

\* @throws IOException if an error occurs while writing to the file

\*/

public void writeResultToFile(double result, String filePath) throws IOException {

try (FileWriter writer = new FileWriter(filePath)) {

writer.write("Calculation result: " + result);

}

}

}

**Lab4Petruk.java**

***package KI\_306.Petruk.Lab4;***

*import java.io.IOException;*

*import java.util.InputMismatchException;*

*import java.util.Scanner;*

*/\*\**

*\* Driver class for running the expression calculation program.*

*\* <p>*

*\* The Main class is responsible for initializing the expression calculator and interacting with the user via the console.*

*\* The user inputs the value of x, after which the program calculates the expression and outputs the result.*

*\* The result is also saved to a file.*

*\* </p>*

*\**

*\* @version 1.0*

*\*/*

*public class Lab4PetrukAlex36 {*

*/\*\**

*\* Entry point of the program. Handles user input, calculates the result of the expression,*

*\* and writes the result to a file.*

*\**

*\* @param args command-line arguments*

*\*/*

*public static void main(String[] args) {*

*ExpressionCalculator calculator = new ExpressionCalculator();*

*Scanner scanner = new Scanner(System.in);*

*try {*

*// Getting the value of x from the user*

*System.out.print("Enter the value of x: ");*

*String input = scanner.next();*

*// Validation and calculation*

*double x = calculator.validateAndParse(input);*

*double result = calculator.calculate(x);*

*System.out.println("Result: " + result);*

*// Writing the result to a file*

*String filePath = "result.txt";*

*calculator.writeResultToFile(result, filePath);*

*System.out.println("The result has been written to the file: " + filePath);*

*} catch (IllegalArgumentException e) {*

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

*} catch (InputMismatchException e) {*

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

*} catch (IOException e) {*

*System.out.println("File write error: " + e.getMessage());*

*} finally {*

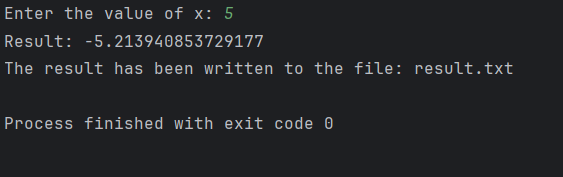
*// Closing the scanner to avoid resource leaks*

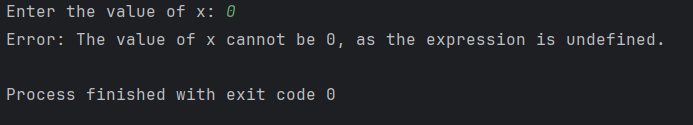
*scanner.close();*

*}*

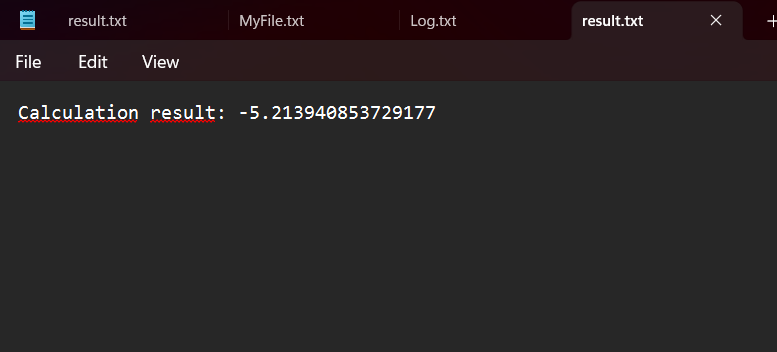
*}*

*}*

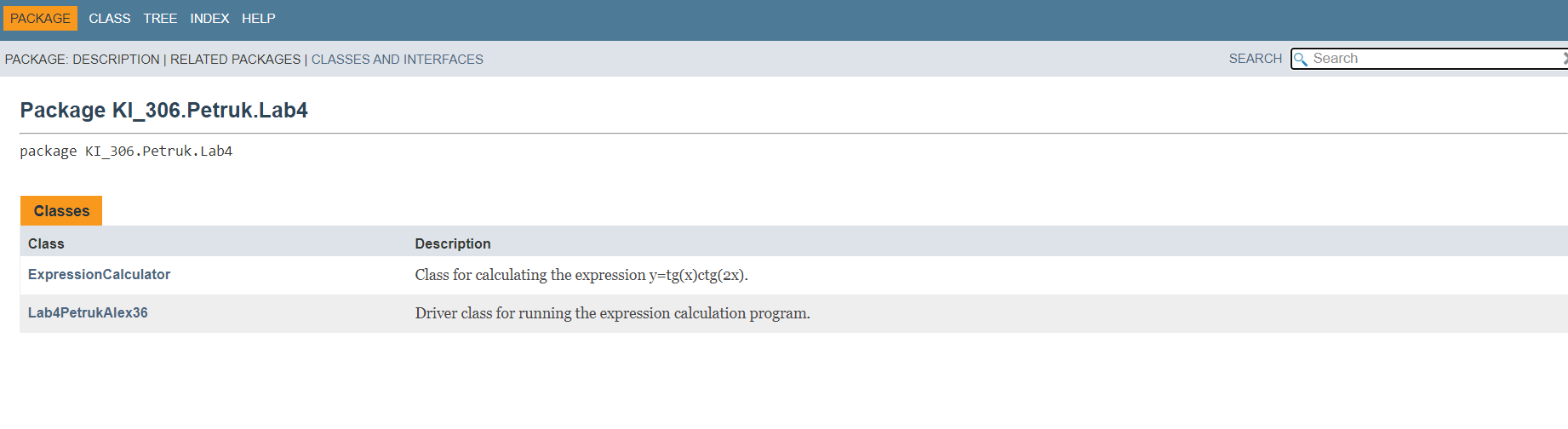




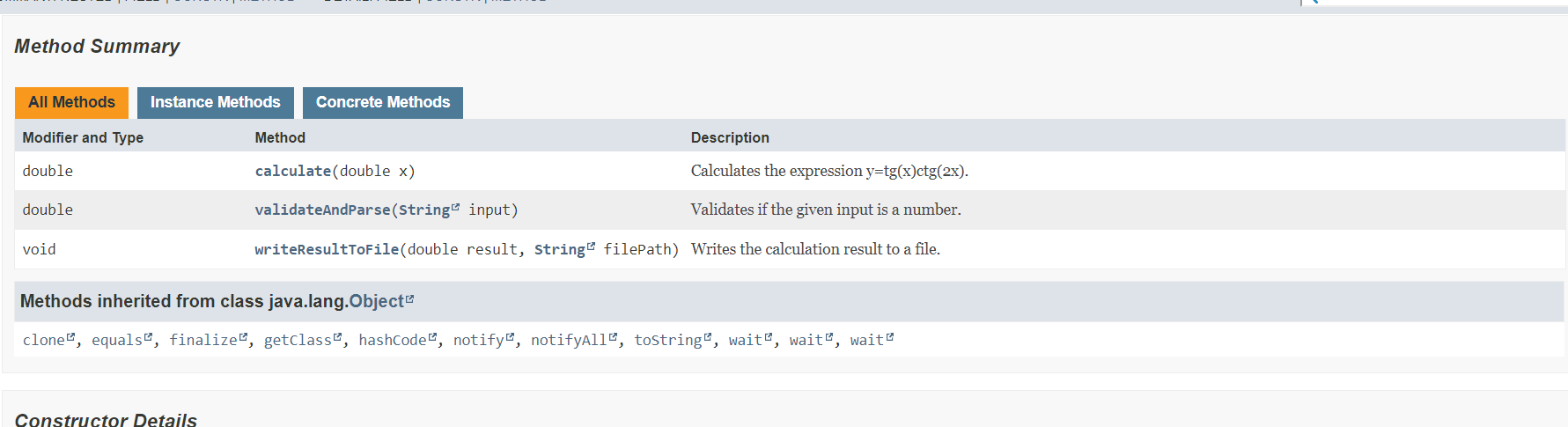
**Рис.1 Вивід результату у консоль**

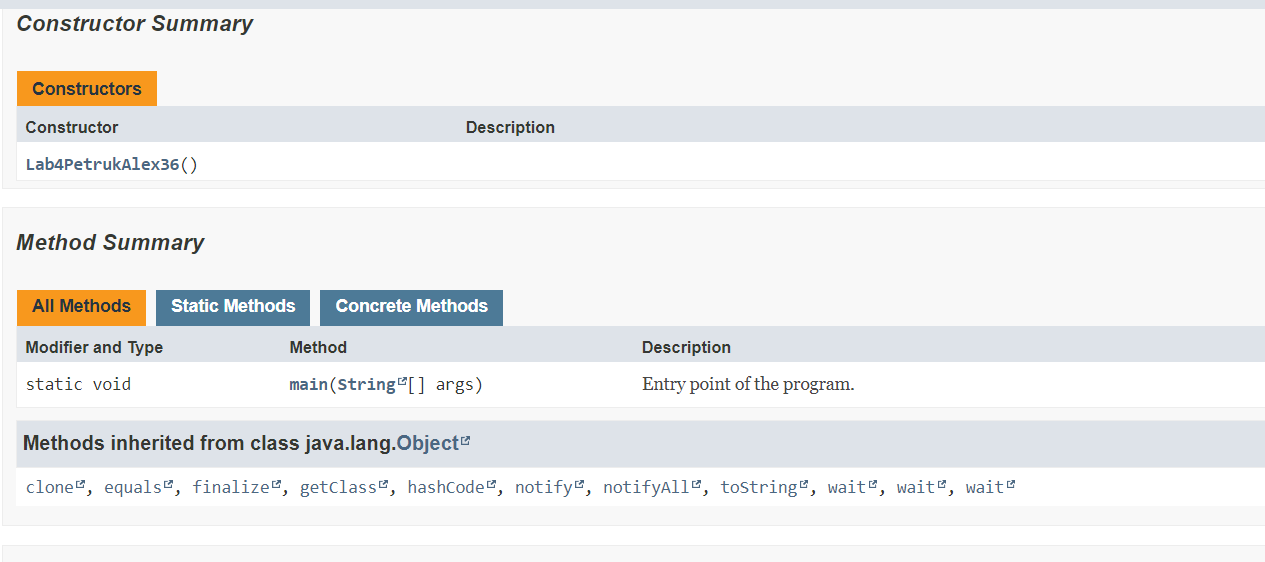


**Рис.2 Вивід результату у текстовий файл**



**Рис.3.1 Фрагмент згенерованої документації**





**Рис.3.2 Фрагмент згенерованої документації**

**Висновок:** На лабораторній роботі я оволодів навиками використання механізму виключень при написанні програм мовою Java.