Міністерство освіти і науки України

Національний університет «Львівська політехніка»



ЗВІТ

Про виконання лабораторної роботи № 5

З дисципліни “Кросплатформні засоби програмування ”

Варіант-25

**Виконала:**

студ. групи КІ-35

Янчинська Н. Р.

**Прийняв:**

Іванов Ю. С.

Львів 2022

ВИКЛЮЧЕННЯ

Мета: оволодіти навиками використання механізму виключень при написанні програм мовою 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.Lab5 та володіти коментарями, які дозволять автоматично згенерувати документацію до розробленогопакету.

2. Автоматично згенерувати документацію до розробленого пакету.

3. Скласти звіт про виконану роботу з приведенням тексту програми, результату її виконання та фрагменту згенерованої документації.

4. Дати відповідь на контрольні запитання.

ВАРІАНТИ ЗАВДАННЬ

25. y=1/sin(x)

Код програми:

Main.java

package KI35.Yanchynska.Lab5;

import java.util.Scanner;

import java.io.\*;

import static java.lang.System.out;

public class Main {

public static void main(String[] args)

{

try

{

out.print("Enter file name: ");

Scanner in = new Scanner(System.in);

String fName = in.nextLine();

PrintWriter fout = new PrintWriter(new File(fName));

try

{

try

{

Equations eq = new Equations();

out.print("Enter X: ");

fout.print(eq.calculate(in.nextInt()));

}

finally

{

fout.flush();

fout.close();

}

}

catch (CalcException ex)

{

out.print(ex.getMessage());

}

}

catch (FileNotFoundException ex)

{

out.print("Exception reason: Perhaps wrong file path");

}

catch (NullPointerException ex)

{

out.print("Exception reason: String can not be null!");

}

}

}

Equations.java

package KI35.Yanchynska.Lab5;

public class Equations {

public double calculate(int x) throws CalcException

{

double y, rad;

rad = x \* Math.PI / 180.0;

try

{

y = 1 / Math.sin(rad);

if (y==Double.NaN || y==Double.NEGATIVE\_INFINITY || y==Double.POSITIVE\_INFINITY || x==90 || x== -90)

throw new ArithmeticException();

else if(y == 0) {

throw new NullPointerException();

}

}

catch (ArithmeticException | NullPointerException ex)

{

if (rad==Math.PI/2.0 || rad==-Math.PI/2.0)

throw new CalcException("Exception reason: Illegal value of X for tangent calculation");

else if (x==0)

throw new CalcException("Exception reason: X = 0");

else

throw new CalcException("Unknown reason of the exception during exception calculation");

}

return y;

}

}

CalcException.java

package KI35.Yanchynska.Lab5;

class CalcException extends ArithmeticException{

public CalcException(){}

public CalcException(String cause)

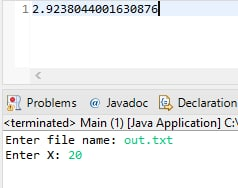
{

super(cause);

}

}

Результат виконання програми:



КОНТРОЛЬНІ ПИТАННЯ

Дайте визначення терміну «виключення».

Виключення – це механізм мови Java, що забезпечує негайну передачу керування блоку коду опрацювання критичних помилок при їх виникненні уникаючи процесу розкручування стеку.

Як створити власний клас виключень?

Приклад власного класу виключень:

class FileFormatException extends IOException

{

public FileFormatException()

{}

public FileFormatException(String message)

{

/\*

Даний виклик конструктора суперкласу дозволяє використовувати метод toString() класу Throwable

\*/

super(message);

}

}

Висновок:

На цій лабораторній роботі я ознайомилася з навиками використання механізму виключень при написанні програм мовою Java.