

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Кафедра «Електронних обчислювальних машин»



Звіт
з лабораторної роботи № 5
з дисципліни: «Кросплатформенні засоби програмування»
на тему: «ВИКЛЮЧЕННЯ »

Виконав:
студент групи КІ-35
Луцюз М.Ю.
Прийняв:
доцент кафедри ЕОМ
Іванов Ю. С.

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 розробленого пакету.
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.

18. $y = \frac{\text{tg}(x)}{(\sin(4x) - 2\cos(x))}$

$$y = \frac{\text{tg}(x)}{\sin(4x) - 2 * \cos(x)}$$

Текст програми

Main.java

```
import static java.lang.System.out;
import java.util.Scanner;
import java.io.*;
/**
 *
 * @author Maksym Lutsiuk
 * @version 1.0
 */
public class Main {
    public static void main(String[] args) {
        {
            try {

                Scanner in = new Scanner(System.in);
                String fName = "Lab_5_Lutsiuk.txt";
                PrintWriter fout = new PrintWriter((fName));
                try {

                    out.print("Enter X: ");
                    Math_Class eq = new Math_Class(in.nextInt());
                    fout.print(eq.calculate());
                } finally {

                    fout.flush();
                }
            }
        }
    }
}
```

```

        fout.close();
    }
} catch (CalcException | ArithmeticException ex) {

    out.print(ex.getMessage());
}
} catch (FileNotFoundException ex) {
    out.print("Exception reason: Perhaps wrong file path");
}

}

}

```

MathClass.java

```
public class Math_Class {

    private double x,y,rad;

    public Math_Class() {

    }

    public Math_Class(double x){
        this.x = x;
        rad = x * Math.PI / 180.0;
    }

    public double calculate() throws CalcException {
        try{
            y = Math.tan(x)/(Math.sin(4*x) - 2 * Math.cos(x) );

            if (y==Double.NaN || y==Double.NEGATIVE_INFINITY ||
                y==Double.POSITIVE_INFINITY || x==90 || x== -90)
                throw new ArithmeticException();
        }
        catch (ArithmeticException ex)
        {
            if(x == rad/2 || x == (3*rad)/2||(Math.sin(4*x) - 2 *
Math.cos(x) == 0))

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

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

        System.out.println(y);
        return y;
    }
}
```

CalcException.java

```
public class CalcException extends Throwable {
    public CalcException(){}
    public CalcException(String cause)
    {
```

```
        super(cause);  
    }  
  
}
```

Результат роботи програми

```
0.4986267123353101|
```

Фрагмент згенерованої документації

All Classes and Interfaces	
All Classes and Interfaces	Exceptions
Class	Description
CalcException	
Equation	Class Equation check exception and calc
EquationsApp	Class EquationsApp test class

```
public class EquationsApp
extends ObjectⓂ
```

Class EquationsApp test class

Constructor Summary

Constructors	
Constructor	Description
EquationsApp()	

Method Summary

All Methods	Static Methods	Concrete Methods
Modifier and Type	Method	Description
static void	main(String [Ⓜ] [] args)	

Methods inherited from class java.lang.Object[Ⓜ]

clone[Ⓜ], equals[Ⓜ], finalize[Ⓜ], getClass[Ⓜ], hashCode[Ⓜ], notify[Ⓜ], notifyAll[Ⓜ], toString[Ⓜ], wait[Ⓜ], wait[Ⓜ], wait[Ⓜ]

Контрольні питання

1. 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

2. У яких ситуаціях використання виключень є виправданим?

Генерація виключень застосовується при:

- помилках введення, наприклад, при введенні назви неіснуючого файлу або Інтернет адреси з подальшим зверненням до цих ресурсів, що призводить до генерації помилки системним програмним забезпеченням;
- збоях обладнання;
- помилках, що пов'язані з фізичними обмеженнями комп'ютерної системи, наприклад, при заповненні оперативної пам'яті або жорсткого диску;
- помилках програмування, наприклад, при некоректній роботі методу, читанні елементів порожнього стеку, виходу за межі масиву тощо.

Висновок: на даній лабораторній роботі я ознайомився з поняттям виключення. Самостійно написав виключення до програми та протестував його. Навчився обробляти помилки на виключеннях. Дізнався про структуру синтаксису Java try – catch. На практиці укріпив знання.