Міністерство освіти та науки України

Національний університет «Львівська політехніка»

Кафедра «Електронних обчислювальних машин»



**ЗВІТ**

**з лабораторної роботи № 5**

**з дисципліни:** «Кросплатформенні засоби програмування»

**на тему:** «ВИКЛЮЧЕННЯ»

Виконав: ст. гр. КІ-35

Гоцій М.М.

Прийняв:

доцент кафедри ЕОМ

Іванов Ю.С

Львів-2022

**Мета роботи:** 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.

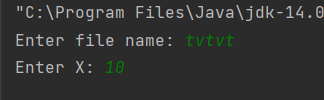
**Варіант 11:**

****

**Текст програми:**

|  |
| --- |
| ***Лістинг CalcException***  package KI35.Hotsii.Lab5;  */\*\*  \* Class <code>CalcException</code> more precises ArithmeticException  \** ***@author*** *Mykyta Hotsii  \** ***@version*** *1.0  \*/* public class CalcException extends ArithmeticException{   public CalcException() {}   public CalcException(String message) {  super(message);  } }  ***Лістинг Accountant***  package KI35.Hotsii.Lab5;  */\*\*  \* Class <code>Accountant</code> implements method for tg(x)/ctg(x) expression  \* calculation  \** ***@author*** *Mykyta Hotsii  \** ***@version*** *1.0  \*/* public class Accountant {  private double rad;   */\*\*  \* Method calculates the tg(x)/ctg(x) expression  \** ***@param*** *x - angle in degrees  \** ***@throws*** *CalcException  \*/* public double calculate(int x) throws CalcException{  double result;  try {  rad = x \* Math.*PI* / 180;  result = Math.*pow*(Math.*tan*(rad), 2);   if (result == Double.*POSITIVE\_INFINITY* || result == Double.*NEGATIVE\_INFINITY*) {  throw new ArithmeticException();  }  } catch (ArithmeticException e) {   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 result;  } }  ***Лістинг Main***  import KI35.Hotsii.Lab5.Accountant; import KI35.Hotsii.Lab5.CalcException;  import java.io.File; import java.io.FileNotFoundException; import java.io.PrintWriter; import java.util.Scanner;  */\*\*  \* Class <code>Accountant</code> Implements driver for Accountant class  \** ***@author*** *Mykyta Hotsii  \** ***@version*** *1.0  \*/* public class Main {  */\*\*  \** ***@param*** *args  \*/* public static void main(String[] args)  {  try  {  System.*out*.print("Enter file name: ");  Scanner in = new Scanner(System.*in*);  String fName = in.nextLine();  PrintWriter fout = new PrintWriter(new File(fName));  try  {  try  {  Accountant eq = new Accountant();  System.*out*.print("Enter X: ");  fout.print(eq.calculate(in.nextInt()));  }  finally  {  fout.flush();  fout.close();  }  }  catch (CalcException ex)  {  System.*out*.print(ex.getMessage());  }  }  catch (FileNotFoundException ex)  {   System.*out*.print("Exception reason: Perhaps wrong file path");  }  } } |

**Результат роботи:**

****

****

**Відповіді на контрольні запитання:**

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.

1. 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:**

Під час виконання роботи я оволодів навиками використання механізму виключень при написанні програм мовою Java.