Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра «Електронних обчислювальних машин»



Звіт

з лабораторної роботи № 6

з дисципліни: «Кросплатформенні засоби програмування»

на тему: «ФАЙЛИ »

**Виконав:**

студент групи КІ-35

Луцюк М.Ю.

**Прийняв:**

доцент кафедри ЕОМ

Іванов Ю. С.

Львів – 2022

Мета: 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 №5. Написати програму для тестування коректності роботи розробленого класу.

2. Для розробленої програми згенерувати документацію.

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

4. Дати відповідь на контрольні запитання.



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 fTxtName = "Lab\_6\_Lutsiuk.txt";  
 String fBinName = "Lab\_6\_Lutsiuk.bin";  
 boolean checkerready = false;  
 Writting\_Clas writter = new Writting\_Clas(fTxtName,fBinName);  
 while (!checkerready) {  
  
 *out*.print("Enter X: ");  
 double KeyA = in.nextDouble();  
 if(KeyA == 0){  
 checkerready = true;  
 break;  
 }  
 Math\_Class eq = new Math\_Class(KeyA);  
  
 writter.writeResTxt(KeyA, eq.calculate());  
  
 }  
 writter.EndOfFile();  
  
  
 } catch (CalcException | IOException ex) {  
  
 *out*.print(ex.getMessage());  
 }  
  
  
  
 }  
 }  
}

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");  
 }  
  
  
 return y;  
 }  
}

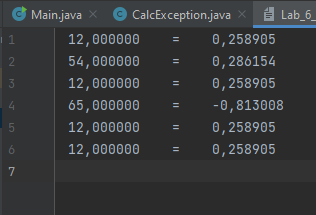
Calc.Exceotion.java

public class CalcException extends ArithmeticException{  
 public CalcException(){}  
  
 public CalcException(String message){  
 super(message);  
 }  
}

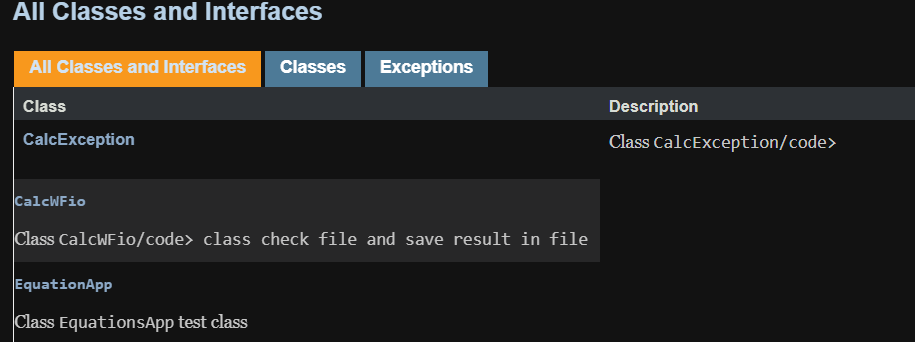
WritingClass.java

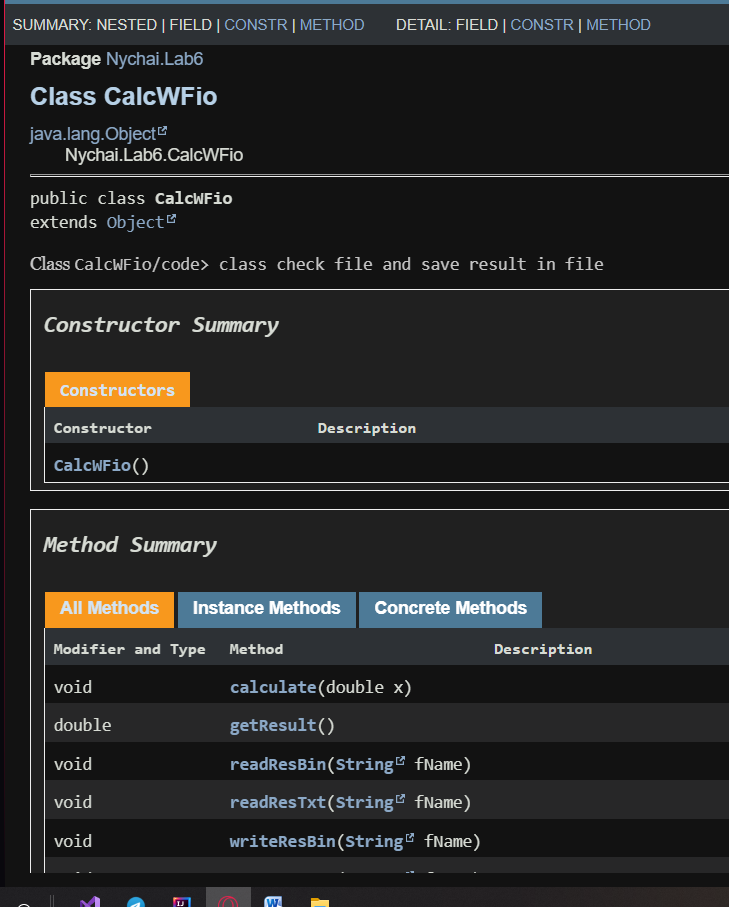
import java.io.\*;  
import java.util.Scanner;  
public class Writting\_Clas {  
 private String FNameTxt;  
 private String FNameBin;  
 DataOutputStream foutBin;  
 PrintWriter foutTxt;  
private double y;  
 public Writting\_Clas( String FNameTxt,String FNameBin) throws FileNotFoundException {  
 this.FNameTxt = FNameTxt;  
 this.FNameBin = FNameBin;  
 foutTxt = new PrintWriter(this.FNameTxt);  
 foutBin = new DataOutputStream(new FileOutputStream(this.FNameBin));  
 }  
  
 public void writeResBin(String fName, double KeyA) throws IOException  
 {  
 foutBin.writeDouble(KeyA);  
 foutBin.writeDouble(y);  
  
 }  
 public void writeResTxt( double KeyA, double KeyB) throws FileNotFoundException  
 {  
  
 foutTxt.printf(" %f = ", KeyA);  
 foutTxt.printf("%f \n",KeyB);  
  
  
 }  
 public void readResTxt()  
 {  
 try  
 {  
 File f = new File (FNameTxt);  
 if (f.exists())  
 {  
 Scanner s = new Scanner(f);  
 System.*out*.println(s.nextLine());  
  
  
 s.close();  
 }  
 else  
 throw new FileNotFoundException("File " + FNameTxt + "not found");  
 }  
 catch (FileNotFoundException ex)  
 {  
 System.*out*.print(ex.getMessage());  
 }  
 }  
 public void readResBin() throws IOException  
 {  
 DataInputStream f = new DataInputStream(new FileInputStream(FNameBin));  
 System.*out*.println(f.readDouble());  
 f.close();  
 }  
 public void EndOfFile() throws IOException {  
 foutTxt.flush();  
 foutTxt.close();  
 foutBin.flush();  
 foutBin.close();  
 }  
}

Результат роботи програми



**Фрагмент згенерованої документації**





Контрольні питання

1. Розкрийте принципи роботи з файловою системою засобами мови Java.

Класи, що спадкуються від абстрактних класів InputStream і OutputStream призначені для здійснення байтового обміну інформацією. Підтримка мовою Java одиниць Unicode, де кожна одиниця має кілька байт, зумовлює необхідність у іншій ієрархії класів, що спадкується від абстрактних класів Reader і Writer. Ці класи дозволяють виконувати операції читання/запису не байтних даних, а двобайтних одиниць Unicode. Принцип здійснення читання/запису даних нічим не відрізняється від такого принципу у інших мовах програмування. Все починається з створення потоку на запис або читання після чого викликаються методи, що здійснюють обмін інформацією. Після завершення обміну даними потоки необхідно закрити щоб звільнити ресурси.

1. 2. Охарактеризуйте клас Scanner.

Для читання текстових потоків найкраще підходить клас Scanner. На відміну від InputStreamReader і FileReader, що дозволяють лише читати текст, він має велику кількість методів, які здатні читати як рядки, так і окремі примітивні типи з подальшим їх перекодуванням до цих типів, робити шаблонний аналіз текстового потоку, здатний працювати без потоку даних та ще багато іншого

Висновок: на даній лабораторній роботі я ознайомився з синтаксисом та принципом роботи з файлами. Покращив запис результату 5 лаб. Практично засвоїв здобуті навички.