

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



**ЗВІТ**  
**з лабораторної роботи № 6**  
**з дисципліни: «Кросплатформенні засоби програмування»**  
**на тему: «ФАЙЛИ»**

Виконав: ст. гр. КІ-35

Дністряк Я. В.

Прийняв:

доцент кафедри ЕОМ

Іванов Ю.С

Львів-2022

**Мета роботи:** оволодіти навиками використання засобів мови Java для роботи з потоками і файлами.

#### ЗАВДАННЯ (4 - варіант)

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:

##### Лістинг FioApp.java

```
package KI35.Dnistrian.Lab6;
import static java.lang.System.out;
import java.io.*;
import java.util.*;
public class FioApp {
    /**
     * @param args
     */
    public static void main(String[] args) throws FileNotFoundException,
    IOException {
        // TODO Auto-generated method stub
        CalcWFio obj = new CalcWFio();
        boolean yes =true;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter data: ");
        double data = s.nextDouble();
        try {
            obj.calculate(data);
        }
        catch(ExceptionDiv0 ex){
            yes=false;
            out.print(ex.getMessage());
        }
        if(yes) {
            System.out.println("\nResult is: " + obj.getResult());
            obj.writeResTxt("textRes.txt");
            obj.writeResBin("BinRes.bin");
            obj.readResBin("BinRes.bin");
            System.out.println("Result is: " + obj.getResult());
            obj.readResTxt("textRes.txt");
            System.out.println("Result is: " + obj.getResult());
        }
    }
}
```

```

* class CalcWfio implements writing and reading data
* @author Slawek
*/
class CalcWfio
{
    public void writeResTxt(String fName) throws FileNotFoundException
    {
        PrintWriter f = new PrintWriter(fName);
        f.printf("%f ", result);
        f.close();
    }

    public void readResTxt(String fName)
    {
        try
        {
            File f = new File (fName);
            if (f.exists())
            {
                Scanner s = new Scanner(f);
                result = s.nextDouble();
                s.close();
            }
            else throw new FileNotFoundException("File " + fName + "not
found");
        }
        catch (FileNotFoundException ex)
        {
            System.out.print(ex.getMessage());
        }
    }

    public void writeResBin(String fName) throws FileNotFoundException,
IOException
    {
        DataOutputStream f = new DataOutputStream(new
FileOutputStream(fName));
        f.writeDouble(result);
        f.close();
    }

    public void readResBin(String fName) throws FileNotFoundException,
IOException
    {
        DataInputStream f = new DataInputStream(new
FileInputStream(fName));
        result = f.readDouble();
        f.close();
    }
}
/**
 * Method calculate the cos(x)/sin(x) expression
 * @param <code>x</code> Angle in degrees
 * @throws CalcException
 */
public void calculate(double x) throws ExceptionDiv0
{
    double y, rad;
    rad = x * Math.PI / 180.0;
    try
    {
        y = Math.cos(rad)/Math.sin(rad);
    }
    // Якщо результат не є числом, то генеруємо виключення
}

```

```

        if (y==Double.NaN || y==Double.NEGATIVE_INFINITY ||
y==Double.POSITIVE_INFINITY || rad % Math.PI==0)
            throw new ArithmeticException();
    }
    catch (ArithmeticException ex)
    {
        // створимо виключення вищого рівня з поясненням причини
        // виникнення помилки
        if (rad % Math.PI==0)
            throw new ExceptionDiv0("Exception reason: cos(x) / 0");
        else
            throw new ExceptionDiv0("Unknown reason of the exception
during exception calculation");
    }
    result=y;
}
/**
 * method returns result
 * @return result
 */
public double getResult()
{
    return result;
}
private double result;
}
}

```

## Лістинг ExceptionDiv0.java

```

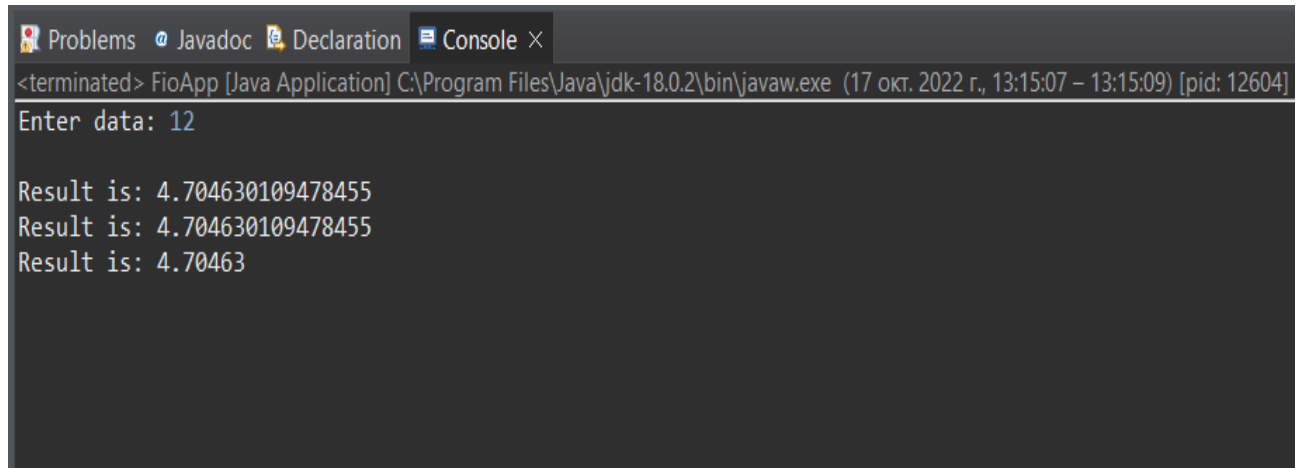
package KI35.Dnistrian.Lab6;

public class ExceptionDiv0 extends ArithmeticException {

    public ExceptionDiv0(){
    }
    public ExceptionDiv0(String cause)
    {
        super(cause);
    }
}

```

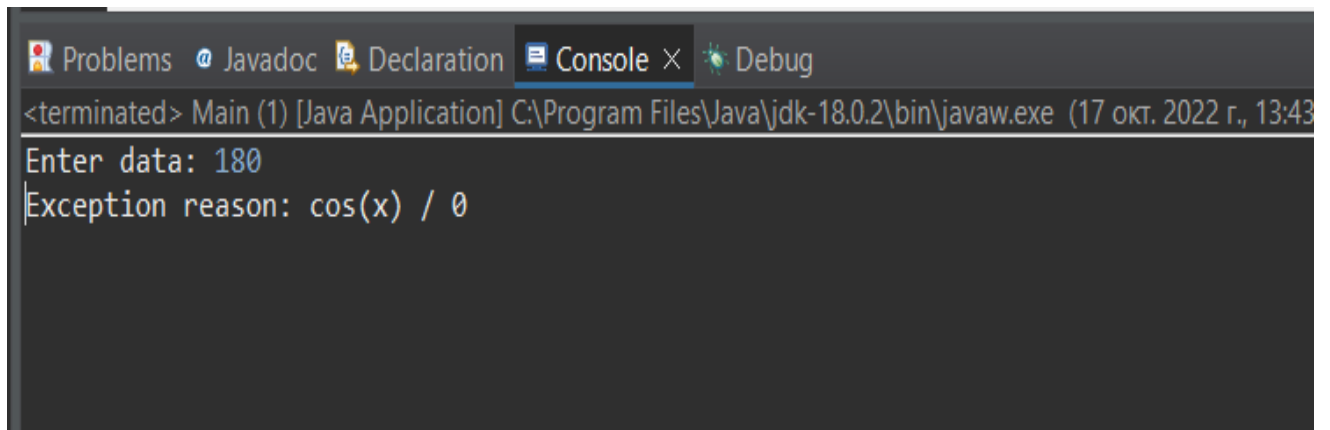
## Результат виконання програми



```
<terminated> FioApp [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe (17 окт. 2022 г., 13:15:07 – 13:15:09) [pid: 12604]
Enter data: 12

Result is: 4.704630109478455
Result is: 4.704630109478455
Result is: 4.70463
```

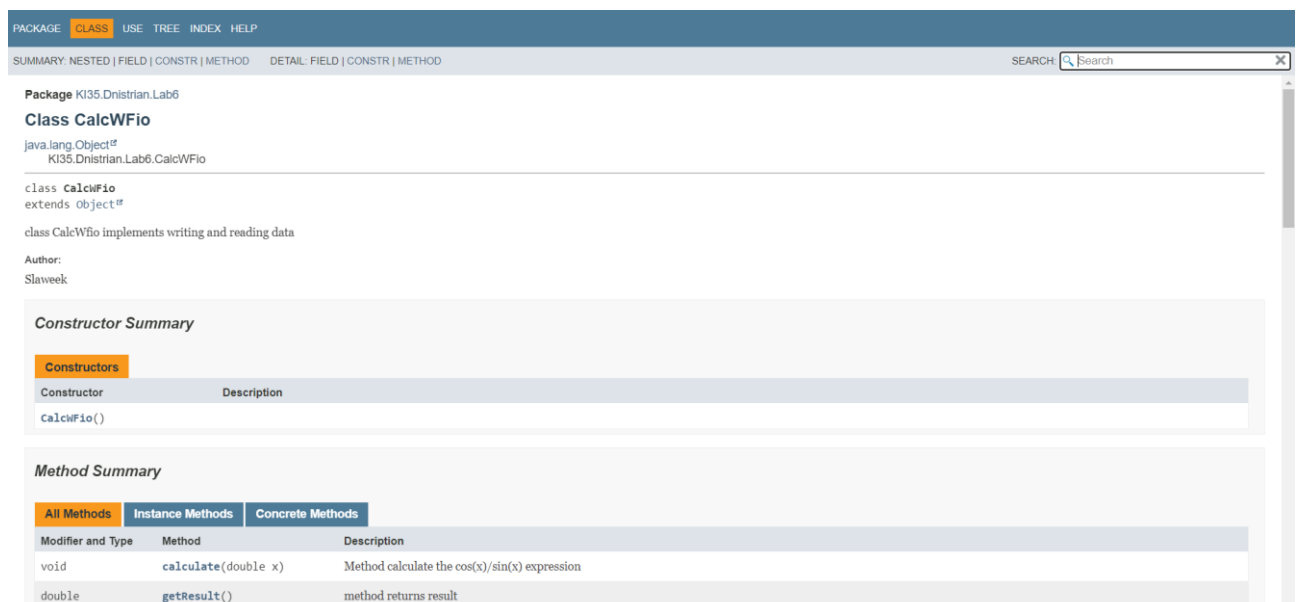
Рис.2.Результати роботи програми



```
<terminated> Main (1) [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe (17 окт. 2022 г., 13:43
Enter data: 180
Exception reason: cos(x) / 0
```

Рис.3. Виключена ситуація

## Фрагмент згенерованої документації



PACKAGE	CLASS	USE	TREE	INDEX	HELP
SUMMARY: NESTED   FIELD   CONSTR   METHOD    DETAIL: FIELD   CONSTR   METHOD    SEARCH: <input type="text"/>					
<b>Package</b> Ki35.Dnistrian.Lab6					
<b>Class CalcWFio</b>					
java.lang.Object <sup>®</sup> Ki35.Dnistrian.Lab6.CalcWFio					
class <b>CalcWFio</b> extends Object <sup>®</sup>					
class CalcWFio implements writing and reading data					
Author: Slaweeek					
<b>Constructor Summary</b>					
<b>Constructors</b>					
Constructor	Description				
CalcWFio()					
<b>Method Summary</b>					
<b>All Methods</b> <b>Instance Methods</b> <b>Concrete Methods</b>					
Modifier and Type	Method	Description			
void	calculate(double x)	Method calculate the cos(x)/sin(x) expression			
double	getResult()	method returns result			

## **Відповіді на контрольні запитання:**

### **1. 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.

### **2. За допомогою якого класу можна здійснити запис у текстовий потік?**

Для буферизованого запису у текстовий потік найкраще використовувати клас PrintWriter. Цей клас має методи для виводу рядків і чисел у текстовому форматі: print, println, printf, - принцип роботи яких співпадає з аналогічними методами System.out.

**Висновок:** виконуючи лабораторну роботу №6 , я оволодів навиками використання засобів мови Java для роботи з потоками і файлами.