

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Кафедра ЕОМ



Звіт
до лабораторної роботи № 7
з дисципліни: «Кросплатформні засоби програмування»
«ФАЙЛИ»
Варіант №22

Виконав:
ст.гр. КІ-34
Трач В.І.
Прийняв:
Іванов Ю.С.

Львів 2022

Мета роботи: 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.

Варіант 22

Код програми :

```
package KI34.Trach.Lab6;
import java.io.*;
import java.util.*;

public class FioApp {
    public static void main(String[] args) throws IOException,
XEqualsFiveException {
        // TODO Auto-generated method stub
        CalcWFio obj = new CalcWFio();
        Scanner s = new Scanner(System.in);
        System.out.print("Enter data: ");
        double data = s.nextDouble();
        obj.calculate(data);
        System.out.println("Result 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());
        // дописати трай кетч блок для обробки експешина у ситуації коли файла
не існує
    }
}
class CalcWFio
{
    public void writeResTxt(String fName)
    {
        try
        {
            File f = new File (fName);
            if (f.exists())
            {
                PrintWriter f32 = new PrintWriter(fName);
                f32.printf("%f ",result);
                f32.close();
            }
            else
                throw new FileNotFoundException("File " + fName + " not found");
        }
        catch (FileNotFoundException ex)
        {
            System.out.print(ex.getMessage());
        }
    }

    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();
}

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

public void calculate(double x) throws CalcException, XEqualsFiveExeption {
    double y, rad;
    if (x == 5){
        throw new XEqualsFiveExeption();
    }
    rad = x * Math.PI / 180.0;
    try
    {
        y = (Math.tan(rad * 4)) / x;

        // Якщо результат не є числом, то генеруємо виключення
        if (Double.isNaN(y) || y==Double.NEGATIVE_INFINITY ||
y==Double.POSITIVE_INFINITY || x == 90 / 4 || x == -90 / 4)
            throw new ArithmeticException();
    }
    catch (ArithmeticException ex)
    {
        // створимо виключення вищого рівня з поясненням причини
        // виникнення помилки
        if (rad * 4 == Math.PI/2.0 || rad * 4 ==- 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");
    }
    result = y;
}

public double getResult()
{

```

```
        return result;
    }
    private double result;
}
```

Результат виконання програми:

Вивід в консоль :

```
Enter data: 2  
Result is: 0.07027041735119573  
Result is: 0.07027041735119573  
Result is: 0.07027
```

Висновок : Я оволодів навиками використання засобів мови Java для роботи з потоками і файлами.