

Міністерство освіти і науки України

Національний університет «Львівська політехніка»



Звіт

З лабораторної роботи №2

З дисципліни «Кросплатформенні засоби програмування»

На тему: «ДОСЛІДЖЕННЯ БАЗОВИХ КОНСТРУКЦІЙ МОВИ JAVA»

Виконала:
ст.гр. КІ-34
Олексій М. В.
Прийняв:
Іванов Ю.С.

Львів-2022

Мета роботи: ознайомитися з базовими конструкціями мови Java та оволодіти навиками написання й автоматичного документування простих консольних програм мовою Java..

ТЕОРЕТИЧНІ ВІДОМОСТІ

Автоматичне документування

При автоматичній генерації документації використовується утиліта `javadoc`, яка аналізує вміст між `/**` і `*/` та на його базі генерує документацію у форматі `*.html`. Коментарі між `/**` і `*/` прийнято починати з описового тексту, за яким слідують дескриптори. Використання дескрипторів полегшує як автоматичну генерацію документації, так і розуміння коду, до якого відноситься коментар. Дескриптор, на відміну від решти коментарів, починається з символу `@` за яким слідує ім'я дескриптора. Оскільки документація генерується у форматі `*.html`, то між `/**` і `*/` допускається розташування `html`-тегів, включаючи рисунки.

Для автоматичної генерації документації між `/**` і `*/` можна розмістити:

- коментарі до класу;
- коментарі до методів;
- коментарі до полів;
- загальні коментарі.

Коментарі до класу мають бути розміщені після директив `import` безпосередньо перед визначенням класу. Найчастіше цей коментар має вигляд одного або кількох коротких речень:

```
/** Об'єкт класу Person описує особу. Особа має властивості: ім'я, прізвище та стать. */
```

Коментарі до методів розташовуються безпосередньо перед методами, які вони описують. Крім дескрипторів загального призначення для коментування методів використовуються дескриптори:

- `@param` змінна опис Цей дескриптор додає в опис методу розділ “parameters”. Опис цього елементу може складатися з кількох рядків та містити `html`-теги. Всі дескриптори `@param`, що відносяться до одного методу слід групувати разом.

- `@return` опис Цей дескриптор додає в опис методу розділ “returns”. Опис цього елементу може складатися з кількох рядків та містити `html`-теги.

- `@throws` опис_класу Цей дескриптор додає в опис методу інформацію про класи об'єкти яких можуть генеруватися при виключних ситуаціях. Відомості про кожен клас слід описувати в окремому дескрипторі `@throws`.

Коментарі до полів (властивостей) застосовуються, якщо поля є загальнодоступними:

```
/** Чоловіча стать */  
public static final int SEX=1;
```

ЗАВДАННЯ

1. Написати та налагодити програму на мові Java згідно варіанту. Програма має задовольняти наступним вимогам:

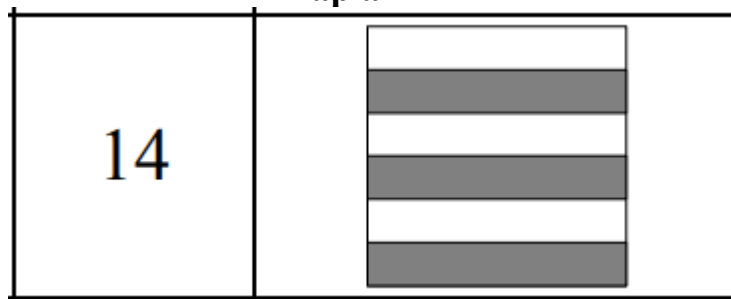
- програма має розміщуватися в загальнодоступному класі Lab2ПрізвищеГрупа;
- програма має генерувати зубчатий масив, який міститиме лише заштриховані області квадратної матриці згідно варіанту;
- розмір квадратної матриці і символ-заповнювач масиву вводяться з клавіатури;
- при не введенні або введенні кількох символів-заповнювачів відбувається коректне переривання роботи програми;
- сформований масив вивести на екран і у текстовий файл;
- програма має володіти коментарями, які дозволять автоматично згенерувати документацію до розробленої програми.

2. Автоматично згенерувати документацію до розробленої програми.

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

4. Дати відповідь на контрольні запитання.

Варіант 14



Код програми:

```
import java.io.*;
import java.util.*;

/**
 * Клас Lab2 реалізує лабораторну роботу №2
 *
 * @author No Information
 * @version 1.0
 * @since version 1.0
 */

public class lab2 {
    /**
```

```

* Статичний метод main є точкою входу в програму
*
* @param args
* @throws FileNotFoundException
*
*/

```

```

public static void main(String[] args) throws FileNotFoundException{

```

```

    int k;
    Scanner in = new Scanner(System.in);
    File dataFile = new File("MyFile.txt");
    PrintWriter fout = new PrintWriter(dataFile);
    System.out.print("Enter matrix size: ");
    k = in.nextInt();
    in.nextLine();

```

```

    char[][] twoDimenArray = new char[k][];
    for(int r=0; r<k; r++) {
        if(r%2==0) {
            twoDimenArray[r] = new char[k];
        }
        else {
            twoDimenArray[r] = new char[0];
        }
    }
}

```

```

String symbol1 ;
System.out.print("Enter a placeholder character: ");
symbol1 = in.nextLine();
//in.nextLine();
if(symbol1.length() > 1) {
    System.out.print("\nToo many placeholder characters! ");
}
else if (symbol1.length() < 1) {
    System.out.print("\nNo placeholder entered!");
}

```

```

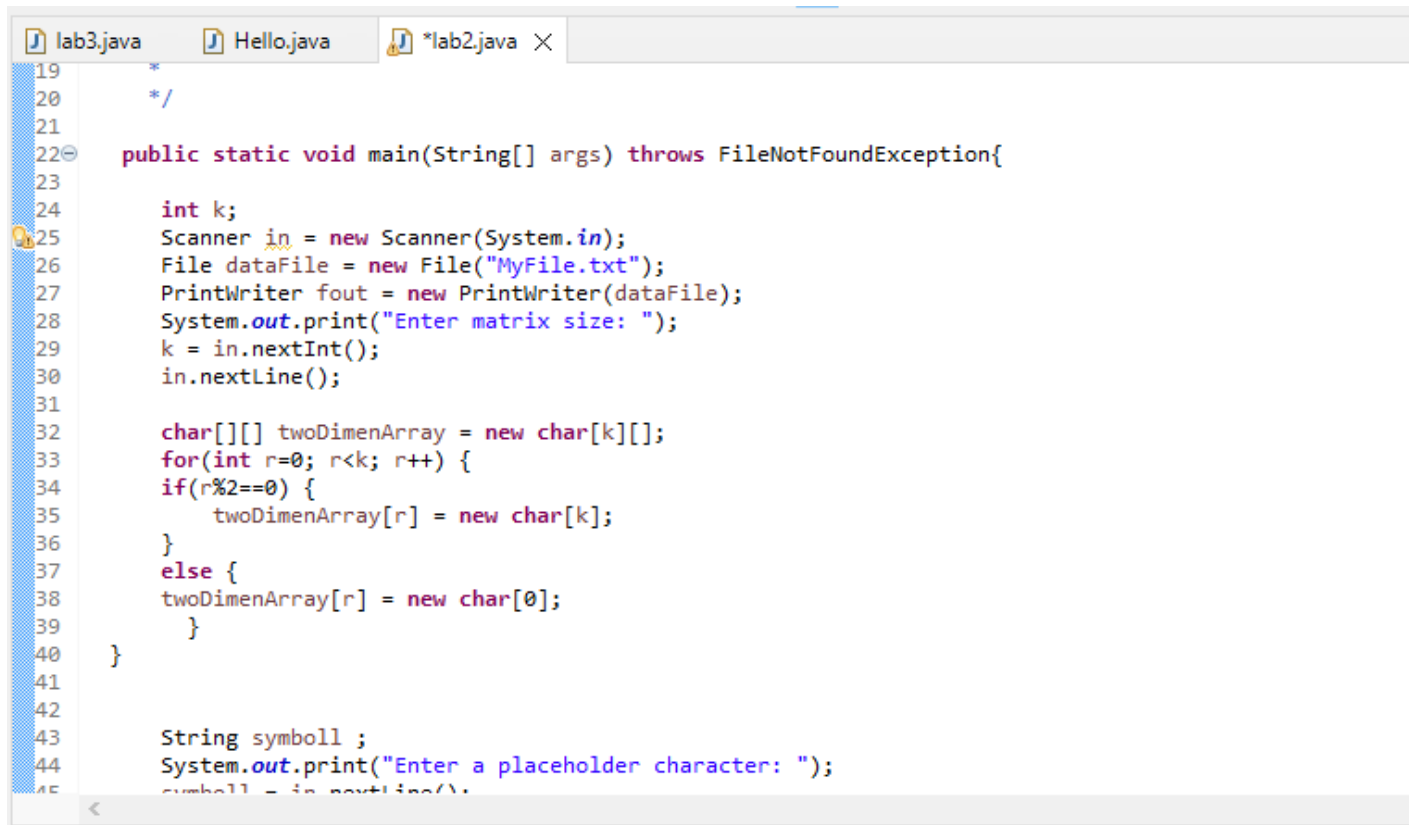
    }
    else {

        //initializing array
        for(int row=0; row < twoDimenArray.length; row++){

            for(int col=0; col < twoDimenArray[row].length; col++){
                twoDimenArray[row][col] = (char) symbol1.codePointAt(0);
            }
        }

        //printing array
        for(int row=0; row < twoDimenArray.length; row++){
            System.out.println();
            for(int col=0; col < twoDimenArray[row].length; col++){
                System.out.print(twoDimenArray[row][col] + " ");
                fout.print(twoDimenArray[row][col] + " ");
            }
        }
        fout.flush();
        fout.close();
    }
}

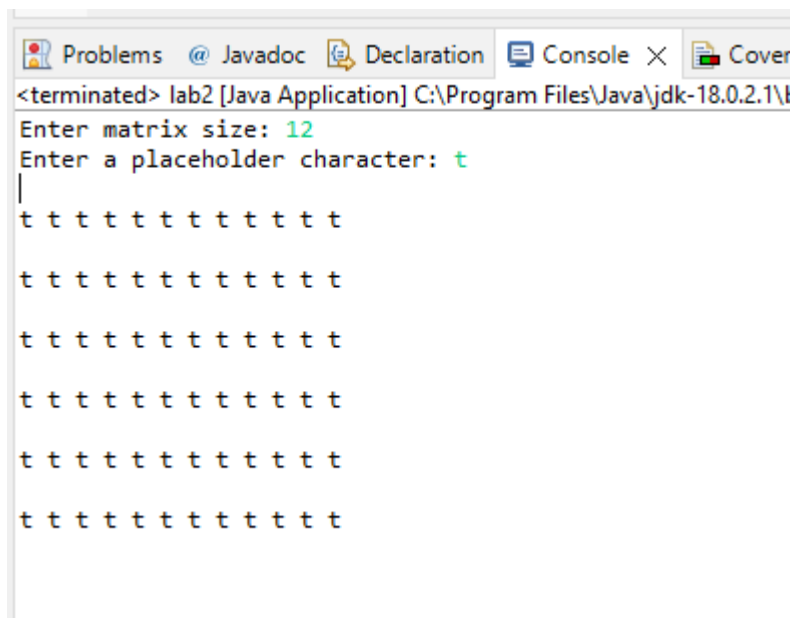
```



```
19  *
20  */
21
22  public static void main(String[] args) throws FileNotFoundException{
23
24      int k;
25      Scanner in = new Scanner(System.in);
26      File dataFile = new File("MyFile.txt");
27      PrintWriter fout = new PrintWriter(dataFile);
28      System.out.print("Enter matrix size: ");
29      k = in.nextInt();
30      in.nextLine();
31
32      char[][] twoDimenArray = new char[k][];
33      for(int r=0; r<k; r++) {
34          if(r%2==0) {
35              twoDimenArray[r] = new char[k];
36          }
37          else {
38              twoDimenArray[r] = new char[0];
39          }
40      }
41
42
43      String symbol ;
44      System.out.print("Enter a placeholder character: ");
45      symbol = in.nextLine();
```

Рис.1. Середовище Eclipse IDE з відкритим проектом Java.

Результат програми:



```
<terminated> lab2 [Java Application] C:\Program Files\Java\jdk-18.0.2.1\
Enter matrix size: 12
Enter a placeholder character: t
|
t t t t t t t t t t t t
t t t t t t t t t t t t
t t t t t t t t t t t t
t t t t t t t t t t t t
t t t t t t t t t t t t
t t t t t t t t t t t t
t t t t t t t t t t t t
```

Висновок: На цій лабораторній роботі ознайомила з базовими конструкціями мови Java та оволоділа навиками написання й автоматичного документування простих консольних програм мовою Java..