

Міністерство освіти і науки України

Національний університет “Львівська політехніка”

Кафедра ЕОМ



Звіт

З лабораторної роботи №6

Варіант – 9

З дисципліни: «Кросплатформні засоби програмування»

На тему: «ПАРАМЕТРИЗОВАНЕ ПРОГРАМУВАННЯ »

Виконав: ст. гр. КІ-305

Заставний Р.А.

Прийняв:

доцент кафедри ЕОМ

Іванов Ю. С.

Львів-2023

Мета роботи: оволодіти навиками параметризованого програмування мовою Java

ЗАВДАННЯ

1. Створити параметризований клас, що реалізує предметну область задану варіантом. Клас має містити мінімум 4 методи опрацювання даних включаючи розміщення та виймання елементів. Парні варіанти реалізують пошук мінімального елементу, непарні – максимального. Написати на мові Java та налагодити програму-драйвер для розробленого класу, яка мстить мінімум 2 різні класи екземпляри яких розмішуються у

екземплярі розробленого класу-контейнеру. Програма має розміщуватися в пакеті Група.Прізвище.Lab6 та володіти коментарями, які дозволять автоматично згенерувати документацію до розробленого пакету.

2. Автоматично згенерувати документацію до розробленого пакету.
3. Завантажити код на GitHub згідно методичних вказівок по роботі з GitHub.
4. Скласти звіт про виконану роботу з приведенням тексту програми, результату її виконання та фрагменту згенерованої документації та завантажити його у ВНС.
5. Дати відповідь на контрольні запитання.

Варіант завдання:

9. Споруда

Код програми:

```
package KI305.Zastavnyi.Lab6;

import java.util.ArrayList;

/**
 * Клас BuildDriver демонструє використання класу Building з різними типами будівель.
 */
public class BuildDriver {
    public static void main(String[] args) {
        Building<? super Data> build = new Building<>();
        build.AddBuild(new School("Гімназія", 166, 17));
        build.AddBuild(new House("Lakeland No. 521, SK S0J 2Y0, Canada", 25));
        build.AddBuild(new House("175 Kingston Ave, Wurtsboro, NY 12790, USA", 24));
        build.AddBuild(new School("Спеціалізована школа I-III ступенів з поглибленим вивчен-
ням англійської мови", 85, 30));
        Data res = build.findMax();
        System.out.print("Найвища будівля: \n");
        res.print();
    }
}

/**
 * Клас Building представляє колекцію будівель типу T.
 *
 * @param <T> Тип будівель, які можуть бути збережені в колекції.
 */
class Building<T extends Data> {
    private ArrayList<T> arr;

    /**
     * Конструктор для ініціалізації ArrayList.
     */
    public Building() {
        arr = new ArrayList<>();
    }
}
```

```

/**
 * Знаходить і повертає будівлю з максимальною висотою.
 *
 * @return Будівля з максимальною висотою.
 */
public T findMax() {
    if (!arr.isEmpty()) {
        T max = arr.get(0);
        for (int i = 1; i < arr.size(); i++) {
            if (arr.get(i).compareTo(max) > 0)
                max = arr.get(i);
        }
        return max;
    }
    return null;
}

/**
 * Додає будівлю типу T до колекції.
 *
 * @param data Будівля для додавання.
 */
public void AddBuild(T data) {
    arr.add(data);
    System.out.print("Будівлю додано: ");
    data.print();
}

/**
 * Видаляє будівлю за вказаним індексом з колекції.
 *
 * @param i Індекс будівлі, яку слід видалити.
 */
public void DeleteData(int i) {
    arr.remove(i);
}
}

/**
 * Інтерфейс Data представляє загальні функції для об'єктів даних з висотою.
 */
interface Data extends Comparable<Data> {
    /**
     * Отримує висоту об'єкта даних.
     *
     * @return Висота об'єкта даних.
     */
    int getHeight();

    /**
     * Виводить інформацію про об'єкт даних.
     */
    void print();
}

/**
 * Клас School представляє школу з ім'ям, номером і висотою.
 */
class School implements Data {
    private String schoolName;
    private int schoolNumber;
    private int height;

    /**

```

```

* Конструктор для ініціалізації об'єкта School.
*
* @param sName Назва школи.
* @param sNumber Номер школи.
* @param sHeight Висота школи.
*/
public School(String sName, int sNumber, int sHeight) {
    schoolName = sName;
    schoolNumber = sNumber;
    height = sHeight;
} public int getHeight()
{
    return height;
}

// Гетери та сетери для властивостей School...

/**
* Порівнює висоту школи з іншим об'єктом Data.
*
* @param p Об'єкт Data для порівняння.
* @return Від'ємне ціле число, нуль або додатнє ціле число, якщо цей об'єкт менший,
рівний або більший за вказаний об'єкт.
*/
public int compareTo(Data p) {
    return Integer.compare(height, p.getHeight());
}

/**
* Виводить інформацію про школу.
*/
public void print() {
    System.out.print("Школа: " + schoolName + ". Номер: " + schoolNumber +
        ". Висота школи: " + height + ";\n");
}
}

/**
* Клас House представляє будинок з адресою і висотою.
*/
class House implements Data {
    private String address;
    private int height;

    /**
    * Конструктор для ініціалізації об'єкта House.
    *
    * @param hAddress Адреса будинку.
    * @param hHeight Висота будинку.
    */
    public House(String hAddress, int hHeight) {
        address = hAddress;
        height = hHeight;
    }

    // Гетери та сетери для властивостей House...

    /**
    * Порівнює висоту будинку з іншим об'єктом Data.
    *
    * @param p Об'єкт Data для порівняння.
    * @return Від'ємне ціле число, нуль або додатнє ціле число, якщо цей об'єкт менший,
    рівний або більший за вказаний об'єкт.
    */

```

```

public int compareTo(Data p) {
    return Integer.compare(height, p.getHeight());
}
public int getHeight()
{
    return height;
}
/**
 * Виводить інформацію про будинок.
 */

public void print() {
    System.out.print("Адреса будинку: " + address + ". Висота будинку: " + height + ";\n");
}
}

```

Результата роботи програми:

```

"C:\Program Files\Microsoft\jdk-17.0.8-hotspot\bin\java.exe" "-javaagent:E:\Programs\IntelliJ IDEA 2023.2.3\lib\idea_rt.jar=57259:
Будівлю додано: Школа: Гімназія. Номер: 166. Висота школи: 17;
Будівлю додано: Адреса будинку: Lakeland No. 521, SK S0J 2Y0, Canada. Висота будинку: 25;
Будівлю додано: Адреса будинку: 175 Kingston Ave, Wurtsboro, NY 12790, USA. Висота будинку: 24;
Будівлю додано: Школа: Спеціалізована школа I-III ступенів з поглибленим вивченням англійської мови. Номер: 85. Висота школи: 30;
Найвища будівля:
Школа: Спеціалізована школа I-III ступенів з поглибленим вивченням англійської мови. Номер: 85. Висота школи: 30;

Process finished with exit code 0

```

Фрагмент згенерованої документації

```

Administrator: Командний рядок

C:\Users\fynti\Desktop\K3П\Lab6ZastavnyiKI305>cd src
C:\Users\fynti\Desktop\K3П\Lab6ZastavnyiKI305\src>cd KI305
C:\Users\fynti\Desktop\K3П\Lab6ZastavnyiKI305\src\KI305>cd Zastavnyi
C:\Users\fynti\Desktop\K3П\Lab6ZastavnyiKI305\src\KI305\Zastavnyi>cd Lab6
C:\Users\fynti\Desktop\K3П\Lab6ZastavnyiKI305\src\KI305\Zastavnyi\Lab6>javadoc BuildDriver.java
Loading source file BuildDriver.java...
Constructing Javadoc information...
Building index for all the packages and classes...
Standard Doclet version 17.0.8.1+1-LTS
Building tree for all the packages and classes...
Generating .\KI305\Zastavnyi\Lab6\BuildDriver.html...
BuildDriver.java:9: warning: no comment
    public static void main(String[] args) {
                        ^
Generating .\KI305\Zastavnyi\Lab6\package-summary.html...
Generating .\KI305\Zastavnyi\Lab6\package-tree.html...
Generating .\overview-tree.html...
Building index for all classes...
Generating .\allclasses-index.html...
Generating .\allpackages-index.html...
Generating .\index-all.html...
Generating .\index.html...
Generating .\help-doc.html...
1 warning
C:\Users\fynti\Desktop\K3П\Lab6ZastavnyiKI305\src\KI305\Zastavnyi\Lab6>

```

Висновок: на цій лабораторній роботі я оволодів навиками параметризованого програмування мовою Java