Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра «Електронних обчислювальних машин»



Звіт

з лабораторної роботи №7

з дисципліни: «Кросплатформенні засоби програмування»

на тему: «Параметризоване програмування»

**Виконав:**

студент групи КІ-35

Береговський І. С.

**Прийняв:**

доцент кафедри ЕОМ

Іванов Ю. С.

Львів – 2022

**Мета роботи:** оволодіти навиками параметризованого програмування мовою Java.

**Завдання (варіант № 28)**

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 9 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. Скласти звіт про виконану роботу з приведенням тексту програми, результату її виконання та фрагменту згенерованої документації. 4. Дати відповідь на контрольні запитання: • Дайте визначення терміну «параметризоване програмування». • Розкрийте синтаксис визначення простого параметризованого класу. • Розкрийте синтаксис створення об’єкту параметризованого класу. • Розкрийте синтаксис визначення параметризованого методу. • Розкрийте синтаксис виклику параметризованого методу. • Яку роль відіграє встановлення обмежень для змінних типів? • Як встановити обмеження для змінних типів? • Розкрийте правила спадкування параметризованих типів. • Яке призначення підстановочних типів? • Застосування підстановочних типів.



**Текст програми**

**Data.java**

package ki\_35.berehovskyi.lab7;

interface Data extends Comparable<Data> {

public int getWeight();

public void print();

}

**Entresol.java**

package ki\_35.berehovskyi.lab7;

import java.util.ArrayList;

public class Entresol <T extends Data> {

private ArrayList<T> arr;

public Entresol()

{

arr = new ArrayList<T>();

}

public T findMin()

{

if (!arr.isEmpty())

{

T min = arr.get(0);

for (int i=1; i< arr.size(); i++)

{

if ( arr.get(i).compareTo(min) < 0 )

min = arr.get(i);

}

return min;

}

return null;

}

public void AddData(T data)

{

arr.add(data);

System.out.print("Element added: ");

data.print();

}

public void DeleteData(int i)

{

arr.remove(i);

}

}

**EntresolDriver.java**

package ki\_35.berehovskyi.lab7;

public class EntresolDriver {

public static void main(String[] args) {

Entresol <? super Data> entresol = new Entresol <Data>();

entresol.AddData(new Pillows("Brinkhaus", 850));

entresol.AddData(new StuffedAnimals("Teddy Bear" , 669));

entresol.AddData(new StuffedAnimals("Unicorn Boo" , 200));

entresol.AddData(new Pillows("WELLPUR" , 420));

Data res = entresol.findMin();

System.out.print("The lightest item in the entresol is: \n");

res.print();

}

}

**Pillows.java**

package ki\_35.berehovskyi.lab7;

public class Pillows implements Data {

private String brandName;

private int pillowsWeight;

public Pillows(String brandName, int pillowsWeight){

this.brandName = brandName;

this.pillowsWeight = pillowsWeight;

}

public String getName()

{

return brandName;

}

public void setName(String brandName)

{

this.brandName = brandName;

}

public int getWeight() {

return pillowsWeight;

}

public int compareTo(Data p)

{

Integer s = pillowsWeight;

return s.compareTo(p.getWeight());

}

public void print() {

System.out.print("Brand of pillow: " + brandName + ", Weight of pillow: " + pillowsWeight + ";\n");

}

}

**Pillows.java**

package ki\_35.berehovskyi.lab7;

public class StuffedAnimals implements Data {

private String animalName;

private int toysWeight;

public StuffedAnimals(String animalName, int toysWeight){

this.animalName = animalName;

this.toysWeight = toysWeight;

}

public String getName()

{

return animalName;

}

public void setName(String animalName)

{

this.animalName = animalName;

}

public int getWeight() {

return toysWeight;

}

public int compareTo(Data p) {

Integer s = toysWeight;

return s.compareTo(p.getWeight());

}

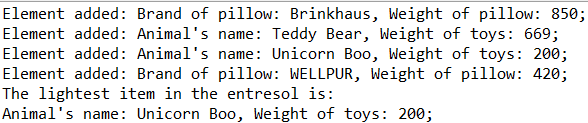
public void print() {

System.out.print("Animal's name: " + animalName + ", Weight of toys: " + toysWeight + ";\n");

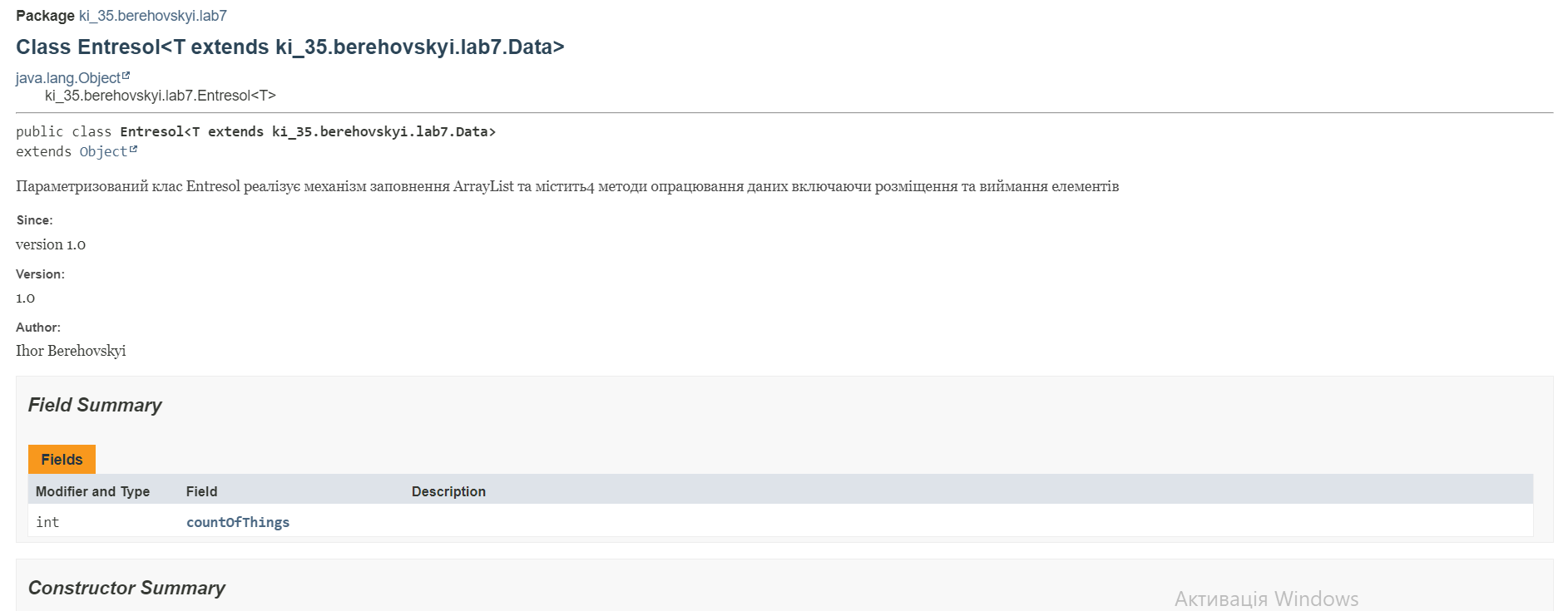
}

}

**Результат виконання програми**



**Фрагмент згенерованої документації**



**Відповіді на контрольні запитання**

• Дайте визначення терміну «параметризоване програмування»..

Параметризоване програмування полягає у написанні коду, що можна багаторазово застосовувати з об’єктами різних класів.

**Висновок**

На даній лабораторній роботі я створив параметризований клас, що реалізує предметну область задану варіантом та містить 4 методи опрацювання даних включаючи розміщення та виймання елементів. Реалізував пошук мінімального елементу за допомогою сompareTo, що дало можливість практично використати даний метод. Отже, я оволодів навичками параметризованого програмування мовою Java.