Міністерство освіти і науки України

Національний університет „Львівська політехніка”

Кафедра ЕОМ



**Звіт**

з лабораторної роботи №7

з дисципліни: “Кросплатформні засоби програмування”

на тему: “Параметризоване програмування”

Виконав: ст. гр. КІ-35

Хван Д.В.

Прийняв:

доцент кафедри ЕОМ

Іванов Ю.С

Львів – 2022

**Мета:** оволодіти навиками параметризованого програмування мовоюJava.

**Індивідуальне завдання:** створити параметризований клас, що реалізує предметну область задану варіантом:

**23. Хлів**

**Хід роботи:**

1. Запустив середовище Visual Code та написав програму згідно індивідуального завдання:

public class BarnApp  
{  
 */\*\*  
 \** ***@param*** *args param  
 \*/* public static void main(String[] args) {  
 Barn <? super Data> Barn = new Barn <Data>();  
 Barn.AddData(new Animal("Pig", 1, 150));  
 Barn.AddData(new Grain("Wheat" , 198));  
 Barn.AddData(new Grain("Oat" , 68));  
 Barn.AddData(new Animal("Chicken" , 1, 5));  
 Data res = Barn.findMax();  
 System.*out*.print("The biggest data in barn is: \n");  
 res.print();  
 }  
}

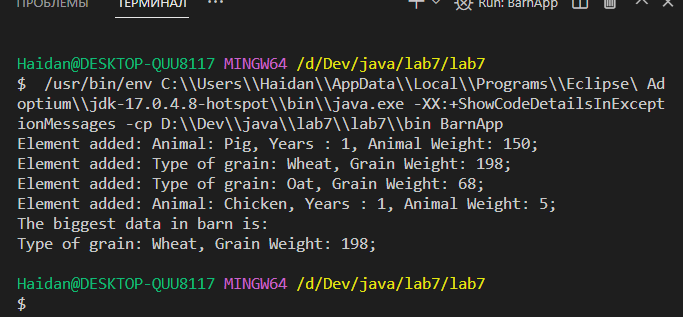
public interface Data extends Comparable<Data>{  
 public int getWeight();  
 public void print();  
  
}

public class Grain implements Data {  
 private String GrainType;  
 private int weight;  
 */\*\*  
 \* Constructor  
 \** ***@param*** *grainType type of grain  
 \** ***@param*** *Weight the weight from all grain's type  
 \*/* public Grain(String grainType, int Weight) {  
 GrainType = grainType;  
 weight = Weight;  
 }  
 */\*\*  
 \* Method returns grain's type  
 \** ***@return*** *grain's type  
 \*/* public String getGrainType() {  
 return GrainType;  
 }  
 */\*\*  
 \* Method sets the grain's type  
 \** ***@param*** *grainType grain's type  
 \*/* public void setGrainType(String grainType) {  
 GrainType = grainType;  
 }  
 */\*\*  
 \* Method returns grain's weight  
 \** ***@return*** *grain's weight  
 \*/* public int getWeight() {  
 return weight;  
 }  
  
 */\*\*  
 \* Method sets the grain's weight  
 \** ***@param*** *weight grain's type  
 \*/* public void setWeight(int weight) {  
 weight = weight;  
 }  
  
 */\*\*  
 \* Method simulates comparing thing`s weight  
 \*/* public int compareTo(Data p) {  
 Integer j = weight;  
 return j.compareTo(p.getWeight());  
 }  
  
  
 */\*\*  
 \* Method simulates printing info about thing  
 \*/* public void print() {  
 System.*out*.print("Type of grain: " + GrainType + ", Grain Weight: " + weight + ";\n");  
 }  
  
  
}

import java.util.ArrayList;  
  
public class Barn <T extends Data>  
{  
 private ArrayList<T> arr;  
 */\*\*  
 \* Constructor  
 \*/* public Barn() {  
 arr = new ArrayList<T>();  
 }  
  
 */\*\*  
 \* Method simulates finding the largest information  
 \*/* 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;  
 }  
  
 */\*\*  
 \* Method simulates adding data  
 \*/* public void AddData(T data)  
 {  
 arr.add(data);  
 System.*out*.print("Element added: ");  
 data.print();  
 }  
  
 */\*\*  
 \* Method simulates deleting data  
 \*/* public void DeleteData(int i)  
 {  
 arr.remove(i);  
 }  
}

import java.time.Year;  
  
public class Animal implements Data {  
 private String AnimalName;  
 private int years;  
 private int weight;  
  
 */\*\*  
 \* Constructor  
 \** ***@param*** *tAnimalName Name of Animal  
 \** ***@param*** *tYears Animal's age  
 \** ***@param*** *tWeight Animal's weight  
 \*/* public Animal(String tAnimalName, int tYears, int tWeight) {  
 AnimalName = tAnimalName;  
 years = tYears;  
 weight = tWeight;  
 }  
 */\*\*  
 \* Method returns animal's name  
 \** ***@return*** *animal's name  
 \*/* public String getAnimalName() {  
 return AnimalName;  
 }  
 */\*\*  
 \* Method sets the new animal's name  
 \** ***@param*** *animalName animal's name  
 \*/* public void setAnimalName(String animalName) {  
 AnimalName = animalName;  
 }  
 */\*\*  
 \* Method returns animal's age  
 \** ***@return*** *animal's age  
 \*/* public int getYears() {  
 return years;  
 }  
 */\*\*  
 \* Method sets the new animal's weight  
 \** ***@param*** *years animal's age  
 \*/* public void setYears(int years) {  
 this.years = years;  
 }  
 */\*\*  
 \* Method returns animal's weight  
 \** ***@return*** *animal's weight  
 \*/* public int getWeight() {  
 return weight;  
 }  
  
 */\*\*  
 \* Method simulates comparing animal's weight  
 \*/* public int compareTo(Data p) {  
 Integer s = weight;  
 return s.compareTo(p.getWeight());  
 }  
 */\*\*  
 \* Method simulates printing info about animal  
 \*/* public void print() {  
 System.*out*.print("Animal: " + AnimalName + ", Years : " + years +  
 ", Animal Weight: " + weight + ";\n");  
 }

**Результат виконання програми (в консолі середовища розробки):**



**Висновок:**

На даній лабораторній роботі оволодів навиками параметризованого програмування мовоюJava.