**Міністерство освіти і науки України**

**Тернопільський національний технічний університет ім. Івана Пулюя**

Факультет інформаційних систем та програмної інжинерії

Кафедра кібербезпеки

**ЗВІТ**

до лабораторної роботи № 6

з дисципліни «Програмування мовою JAVA»

на тему:

**Застосування класів-колекцій і класів утиліт.**

Виконав студент групи СБ-21

Антонов Роман Олександрович

Перевірив:

Луцків Андрій Мирославович

Тернопіль, 2024

Посилання на репозиторій:

<https://github.com/rom341/JavaLabsTNTU>

Результат виконання:

===============================================

ArrayList Demonstration

Animal{name='Lion', age=0}

Animal{name='Tiger', age=15}

Animal{name='Elephant', age=9}

Animal{name='Giraffe', age=18}

Animal{name='Zebra', age=5}

Animal{name='Monkey', age=16}

Animal{name='Panda', age=13}

Animal{name='Kangaroo', age=6}

Animal{name='Koala', age=18}

Animal{name='Cheetah', age=3}

Animal{name='Lion', age=0}

Animal{name='Tiger', age=15}

Animal{name='Lion', age=0}

Generated ArrayList:

[Animal{name='Lion', age=0}, Animal{name='Tiger', age=15}, Animal{name='Elephant', age=9}, Animal{name='Giraffe', age=18}, Animal{name='Zebra', age=5}, Animal{name='Monkey', age=16}, Animal{name='Panda', age=13}, Animal{name='Kangaroo', age=6}, Animal{name='Koala', age=18}, Animal{name='Cheetah', age=3}, Animal{name='Lion', age=0}, Animal{name='Tiger', age=15}, Animal{name='Lion', age=0}]

===============================================

===============================================

LinkedHashSet Demonstration

Animal{name='Lion', age=0}

Animal{name='Tiger', age=15}

Animal{name='Elephant', age=9}

Animal{name='Giraffe', age=18}

Animal{name='Zebra', age=5}

Animal{name='Monkey', age=16}

Animal{name='Panda', age=13}

Animal{name='Kangaroo', age=6}

Animal{name='Koala', age=18}

Animal{name='Cheetah', age=3}

Generated LinkedHashSet:

[Animal{name='Lion', age=0}, Animal{name='Tiger', age=15}, Animal{name='Elephant', age=9}, Animal{name='Giraffe', age=18}, Animal{name='Zebra', age=5}, Animal{name='Monkey', age=16}, Animal{name='Panda', age=13}, Animal{name='Kangaroo', age=6}, Animal{name='Koala', age=18}, Animal{name='Cheetah', age=3}]

===============================================

===============================================

Sorted ArrayList Demonstration

Animal{name='Lion', age=0}

Animal{name='Lion', age=0}

Animal{name='Lion', age=0}

Animal{name='Cheetah', age=3}

Animal{name='Zebra', age=5}

Animal{name='Kangaroo', age=6}

Animal{name='Elephant', age=9}

Animal{name='Panda', age=13}

Animal{name='Tiger', age=15}

Animal{name='Tiger', age=15}

Animal{name='Monkey', age=16}

Animal{name='Giraffe', age=18}

Animal{name='Koala', age=18}

Sorted ArrayList:

[Animal{name='Lion', age=0}, Animal{name='Lion', age=0}, Animal{name='Lion', age=0}, Animal{name='Cheetah', age=3}, Animal{name='Zebra', age=5}, Animal{name='Kangaroo', age=6}, Animal{name='Elephant', age=9}, Animal{name='Panda', age=13}, Animal{name='Tiger', age=15}, Animal{name='Tiger', age=15}, Animal{name='Monkey', age=16}, Animal{name='Giraffe', age=18}, Animal{name='Koala', age=18}]

===============================================

===============================================

TreeSet Demonstration

Animal{name='Lion', age=0}

Animal{name='Cheetah', age=3}

Animal{name='Zebra', age=5}

Animal{name='Kangaroo', age=6}

Animal{name='Elephant', age=9}

Animal{name='Panda', age=13}

Animal{name='Tiger', age=15}

Animal{name='Monkey', age=16}

Animal{name='Giraffe', age=18}

Generated TreeSet:

[Animal{name='Lion', age=0}, Animal{name='Cheetah', age=3}, Animal{name='Zebra', age=5}, Animal{name='Kangaroo', age=6}, Animal{name='Elephant', age=9}, Animal{name='Panda', age=13}, Animal{name='Tiger', age=15}, Animal{name='Monkey', age=16}, Animal{name='Giraffe', age=18}]

===============================================

===============================================

HashMap Demonstration

Entity added: Key (age): 0 Value (animal): Animal{name='Lion', age=0}

Entity added: Key (age): 15 Value (animal): Animal{name='Tiger', age=15}

Entity added: Key (age): 9 Value (animal): Animal{name='Elephant', age=9}

Entity added: Key (age): 18 Value (animal): Animal{name='Giraffe', age=18}

Entity added: Key (age): 5 Value (animal): Animal{name='Zebra', age=5}

Entity added: Key (age): 16 Value (animal): Animal{name='Monkey', age=16}

Entity added: Key (age): 13 Value (animal): Animal{name='Panda', age=13}

Entity added: Key (age): 6 Value (animal): Animal{name='Kangaroo', age=6}

Entity added: Key (age): 18 Value (animal): Animal{name='Koala', age=18}

Entity added: Key (age): 3 Value (animal): Animal{name='Cheetah', age=3}

Entity added: Key (age): 0 Value (animal): Animal{name='Lion', age=0}

Entity added: Key (age): 15 Value (animal): Animal{name='Tiger', age=15}

Entity added: Key (age): 0 Value (animal): Animal{name='Lion', age=0}

Generated HashMap:

{0=Animal{name='Lion', age=0}, 16=Animal{name='Monkey', age=16}, 18=Animal{name='Koala', age=18}, 3=Animal{name='Cheetah', age=3}, 5=Animal{name='Zebra', age=5}, 6=Animal{name='Kangaroo', age=6}, 9=Animal{name='Elephant', age=9}, 13=Animal{name='Panda', age=13}, 15=Animal{name='Tiger', age=15}}

===============================================

Process finished with exit code 0

Код:

Launch:

package org.example.Lab6\_Updated;  
  
public class Launch {  
 public static void main(String[] args) {  
 Demonstrator demonstrator = new Demonstrator();  
 String delimiter = "===============================================";  
  
 System.*out*.println(delimiter);  
 demonstrator.DemonstrateArrayList();  
 System.*out*.println(delimiter);  
  
 System.*out*.println(delimiter);  
 demonstrator.DemonstrateLinkedHashSet();  
 System.*out*.println(delimiter);  
  
 System.*out*.println(delimiter);  
 demonstrator.DemonstrateSortedArrayList();  
 System.*out*.println(delimiter);  
  
 System.*out*.println(delimiter);  
 demonstrator.DemonstrateTreeSet();  
 System.*out*.println(delimiter);  
  
 System.*out*.println(delimiter);  
 demonstrator.DemonstrateHashMap();  
 System.*out*.println(delimiter);  
 }  
}

package org.example.Lab6\_Updated;  
  
import java.util.\*;  
  
public class Demonstrator {  
 private final List<Animal> animals;  
  
 public Demonstrator() {  
 animals = generateAnimalList();  
 }  
  
 private List<Animal> generateAnimalList() {  
 List<Animal> animalList = new ArrayList<>();  
 String[] names = {"Lion", "Tiger", "Elephant", "Giraffe", "Zebra", "Monkey", "Panda", "Kangaroo", "Koala", "Cheetah"};  
  
 Random random = new Random();  
  
 for (String name : names) {  
 int age = random.nextInt(20);  
 animalList.add(new Animal(name, age));  
 }  
 //inserting dublicates  
 animalList.add(animalList.get(0));  
 animalList.add(animalList.get(1));  
 animalList.add(animalList.get(0));  
 return animalList;  
 }  
  
 public void DemonstrateArrayList() {  
 System.*out*.println("ArrayList Demonstration\n");  
 List<Animal> animalArrayList = new ArrayList<>(animals);  
 for (Animal animal : animalArrayList) {  
 System.*out*.println(animal);  
 }  
 System.*out*.printf("\nGenerated ArrayList: \n%s\n", animalArrayList);  
 }  
  
 public void DemonstrateLinkedHashSet() {  
 System.*out*.println("LinkedHashSet Demonstration\n");  
 Set<Animal> animalLinkedHashSet = new LinkedHashSet<>(animals);  
 for (Animal animal : animalLinkedHashSet) {  
 System.*out*.println(animal);  
 }  
 System.*out*.printf("\nGenerated LinkedHashSet: \n%s\n", animalLinkedHashSet);  
 }  
  
 public void DemonstrateSortedArrayList() {  
 System.*out*.println("Sorted ArrayList Demonstration\n");  
 List<Animal> animalArrayList = new ArrayList<>(animals);  
 Collections.*sort*(animalArrayList);  
 for (Animal animal : animalArrayList) {  
 System.*out*.println(animal);  
 }  
 System.*out*.printf("\nSorted ArrayList: \n%s\n", animalArrayList);  
 }  
  
 public void DemonstrateTreeSet() {  
 System.*out*.println("TreeSet Demonstration\n");  
 Set<Animal> animalTreeSet = new TreeSet<>(animals);  
 for (Animal animal : animalTreeSet) {  
 System.*out*.println(animal);  
 }  
 System.*out*.printf("\nGenerated TreeSet: \n%s\n", animalTreeSet);  
 }  
  
 public void DemonstrateHashMap() {  
 System.*out*.println("HashMap Demonstration\n");  
 Map<Integer, Animal> animalHashMap = new HashMap<>();  
 int id = 1;  
  
 for (Animal animal : animals) {  
 animalHashMap.put(animal.getAge(), animal);  
 System.*out*.printf("Entity added: Key (age): %d \t Value (animal): %s\n", animal.getAge(), animal);  
 }  
 System.*out*.printf("\nGenerated HashMap: \n%s\n", animalHashMap);  
 }  
}

package org.example.Lab6\_Updated;  
  
import java.util.Objects;  
  
public class Animal extends Food implements Comparable<Animal> {  
 private String name;  
 private int age;  
  
 public Animal(String name, int age) {  
 this.name = name;  
 this.age = age;  
 }  
  
 public String getName() {  
 return name;  
 }  
  
 public void setName(String name) {  
 this.name = name;  
 }  
  
 public int getAge() {  
 return age;  
 }  
  
 public void setAge(int age) {  
 this.age = age;  
 }  
  
 @Override  
 public boolean equals(Object o) {  
 if (this == o) return true;  
 if (o == null || getClass() != o.getClass()) return false;  
 Animal animal = (Animal) o;  
 return age == animal.age && Objects.*equals*(name, animal.name);  
 }  
  
 @Override  
 public int hashCode() {  
 return Objects.*hash*(name, age);  
 }  
  
 @Override  
 public String toString() {  
 return "Animal{" +  
 "name='" + name + '\'' +  
 ", age=" + age +  
 '}';  
 }  
  
 @Override  
 public int compareTo(Animal o) {  
 return Integer.*compare*(this.age, o.age);  
 }  
}

package org.example.Lab6\_Updated;  
  
public abstract class Food {  
 private String name;  
 private double mass;  
 private boolean isMeat;  
  
 public Food(){}  
  
 public Food(String name, double mass, boolean isMeat) {  
 this.name = name;  
 this.mass = mass;  
 this.isMeat = isMeat;  
 }  
  
 public boolean isMeat() {  
 return isMeat;  
 }  
  
 public void setMeat(boolean meat) {  
 isMeat = meat;  
 }  
  
 public String getName() {  
 return name;  
 }  
  
 public void setName(String name) {  
 this.name = name;  
 }  
  
 public double getMass() {  
 return mass;  
 }  
  
 public void setMass(double mass) {  
 this.mass = mass;  
 }  
}