Lab 05

Part One: (ArrayList) ★★★★☆

Task 01➔ Write a Java program to create a new ArrayList (of type String), then ask a user to enter some element, and print out the collection after each entry.

Task 02➔ Add to the past program a Search method that look in the in ArrayList and return the element that match

Task 03➔ Add to the past program a Search method Search in ArrayList and return a sub list (in case of same element is duplicated)

In rest of the following tasks, our objective is to create a java program that digitizes the student list and their TP notes:

Task 04➔ Create a student class that contain the following attributes: student id (int and auto generated in the constructor), name, student TP note. This class contain only one method that print student information.

Task 05➔ Create a separate Main class that contain a main method to run our program and a static ArrayList of type Student.

Task 06➔ In the Main class, add the following functionalities and methods:

- Allow a user to enter a student information, create a new student object and then add to the ArrayList.
- Allow a user to search a Student by its name.
- Allow a user to search a set Students that have a similar names.
- Allow a user to enter a student TP note.
- Allow a user get average of all classe.

Note: Solution should have these two files: Main.java, Student.java

A Part of the Main class sample:

```
1  import java.util.ArrayList;
2  import java.util.Scanner;
3
4
5  public class Main {
6      static ArrayList<Student> student_list = new ArrayList();
7
8      public static void add_student(){...20 lines }
28
29     public static void show_all_student(){...12 lines }
41
42     public static void search_one(){...14 lines }
56
57     public static void search_many(){...17 lines }
74
75     public static void compute_average_notes(){...10 lines }
85
86     public static void main(String [] args){
```

Program output

```
run:
Enter (add) to add a new student
Enter (show) to show all students
Enter (search) to search student by its name
Enter (searchAll) to search students
Enter (avg) To compute and display the class average
add
Enter Student name:
amine
Enter Student TP note:
15
The student has been added to the list
do you want to add a new student (y/n)?
n
        Student Id:249046391
        Student Name:amine
        Student TP note:15.0

Enter (add) to add a new student
Enter (show) to show all students
Enter (search) to search student by its name
Enter (searchAll) to search students
Enter (avg) To compute and display the class average
```

Dr. Oussama SERHANE (o.serhane@esi-sba.dz)

Part Two: Inheritance ★★★★☆

Objective: The goal of this exercise is to practice working with classes, objects, and inheritance while reinforcing your understanding of UML class diagrams. You will design and implement a Java program that models a real-life scenario using inheritance and ArrayList.

Instructions: Generate an Exercise Idea Using ChatGPT, Deepseek, or Other LLMs:

Come up with a real-life scenario that can be modeled using inheritance. Your exercise must involve at least four related classes where one class serves as a parent (superclass), and the others inherit from it (subclasses).

Design the UML Class Diagram: Represent your superclass and subclasses, their attributes, methods, and relationships using inheritance in a UML diagram.

Evaluation Criteria:

- Creativity in problem selection
- Correct UML class diagram representation
- Proper implementation of classes and objects
- Clean code structure
- Functional and well-Presented

Deliverables:

1. A document with the problem statement and UML class diagram.
2. A Java program implementing the solution.
3. A short report explaining your design choices and how the classes interact.

Dr. Oussama SERHANE (o.serhane@esi-sba.dz)