Hands-on activity:

Collections & Streams and Generics

I.   Collections & Streams

1. Lists

We want to create a simple application to manipulate a list of product-type objects.

1. Create a Product class containing the attributes: id(long), name(String), price(double);

2. Create a class called ProductManagementApp that will contain a main method (main);

3. Inside the main method, create an ArrayList, then:

    - Add products.

    - Delete a product by index.

    - Display the list of products.

    - Modify a product by index.

    - Search for a product whose name is typed by the user.

2. Maps

Create a Hashmap that stores student grades. The key is a String representing the student's name, and the value is a double representing the grade.

1. Insert student grades.

2. Increase a student's grade.

3. Increase a student's grade.

4. Delete a student's grade.

5. Display the size of the map.

6. Display the average, maximum, and minimum grades.

7. Check if there is a grade equal to 20.

8. After each operation, display the list using the forEach loop with the lambda expression.

3. Sets

Create two HashSet sets named groupA and groupB, containing the names of students in groups A and B.

1. Add student names to each HashSet.
2. Display the intersection of the two HashSets.
3. Display the union of the two HashSets.

II. Generics

1. We want to create a generic class called ***GenericStorage*** that can store elements of any type. The class should have an attribute named elements of type *List* to store the elements. Add the following methods to the ***GenericStorage*** class:

   ✓ *public void addElement(T o)*: allows you to add an element to the collection.
   ✓ *public void removeElement(int index):* allows you to remove a specific element from the collection.
   ✓ *public T getElement(int index):* allows you to retrieve an element at a given position in the collection.
   ✓ *public void removeElement(int index):* allows you to remove a specific element from the collection.
   ✓ *public int getSize():* allows you to obtain the current size of the list.

Create a test class called ***Application*** to test your implementation. Use the ***GenericStorage*** class with different types (for example, *Integer, String, Double*). Perform addition, retrieval, deletion, and verification operations on the generic list.

2. The purpose of this task is to use generics with a collection of objects of type product .
   - Create a Product class with the attributes id, name, brand, price, description, and number in stock.
   - Create a generic interface IMetier that will declare the methods for managing our Product entities. This interface has a generic type T and contains the following methods:
     o *public void add(T o):* which allows you to add an object to the list.
     o *public List<T> getAll():* which returns the list of objects in the form of a list
     o *public T findById(long id):* which returns a product by id.
     o *public void delete(long id)*: which deletes an object by id.
   - Create a ***MetierProduitImpl*** class that implements the ***IMetier*** interface. This class contains an attribute that represents a list of products.
   - Develop an ***Application*** class containing the ***main method*** that offers the user the following menu in a while loop:
     1. Display the list of products.
     2. Search for a product by its id.
     3. Add a new product to the list.
     4. Delete a product by id.
     5. Exit this program.