**PG-DAC AUGUST 24 BATCH**

**MUMBAI CDAC**

**MODULE 2 OOPJ**

**ASSIGNMENT 7**

1)Write a Java program that takes a list of integers as input and returns a list of duplicate integers.

import java.util.\*;

public class DuplicateList {

    public static void duplicateList(List<Integer> list) {

        List<Integer> duplicates = new ArrayList<>();

        for (int i = 0; i < list.size(); i++) {

            Integer iElement = list.get(i);

            for (int j = i + 1; j < list.size(); j++) {

                Integer jElement = list.get(j);

                if (iElement.equals(jElement)) {

                    if (!duplicates.contains(iElement))

                        duplicates.add(iElement);

                    break;

                }

            }

        }

        System.out.println("\nDuplicate Elements are :  ");

        for (Integer dupl : duplicates)

            System.out.print(dupl + "  ");

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        List<Integer> list = new ArrayList<>();

        System.out.print("Enter the number of elements: ");

        int n = sc.nextInt();

        System.out.println("Enter " + n + " elements:");

        for (int i = 0; i < n; i++) {

            list.add(sc.nextInt());

        }

        System.out.print("\nList elements are :  ");

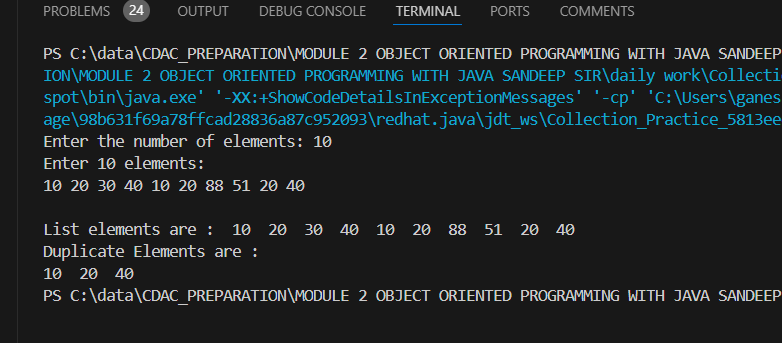
        for (Integer element : list)

            System.out.print(element + "  ");

        duplicateList(list);

    }

}



2)Create a Person class with attributes name and age. Write a Java program that sorts a list of Person objects first by age and then by name if the ages are equal.

import java.util.\*;

class PersonSorting{

    String name;

    int age;

    public PersonSorting() {

    }

    public PersonSorting(String name, int age) {

        this.name = name;

        this.age = age;

    }

    public String toString() {

        return String.format("%-10s%-3d", this.name, this.age);

    }

}

class AgeComparator implements Comparator<PersonSorting>{

    public int compare(PersonSorting p1, PersonSorting p2) {

        return p1.age - p2.age;

    }

}

class NameComparator implements Comparator<PersonSorting>{

    public int compare(PersonSorting p1, PersonSorting p2) {

        return p1.name.compareTo(p2.name);

    }

}

    public class PersonSortingCollection {

    public static void display(Collection<?> list) {

        for (Object p : list) {

            System.out.println(p.toString());

        }

    }

    public static void main(String args[]) {

        List<PersonSorting> list = new ArrayList<>();

        PersonSorting p1 = new PersonSorting("Saurabh", 21);

        PersonSorting p2 = new PersonSorting("Sneha", 23);

        PersonSorting p3 = new PersonSorting("Tushar", 22);

        list.add(p1);

        list.add(p2);

        list.add(p3);

        System.out.println("List of Elements : ");

        for(PersonSorting p: list)

            System.out.printf("%-10s%-3d\n", p.name, p.age);

        System.out.println("\nSort by Age : ");

        Collections.sort(list, new AgeComparator());

        display(list);

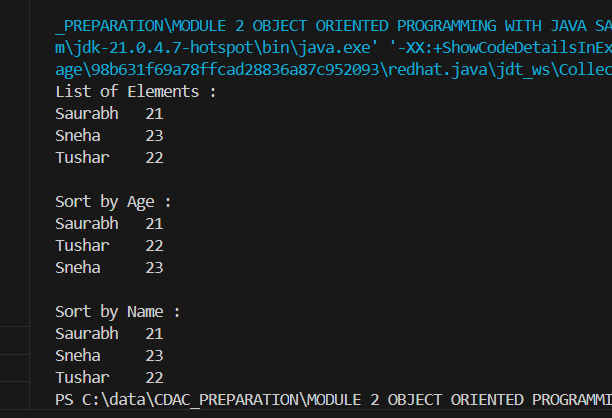
        System.out.println("\nSort by Name : ");

        Collections.sort(list, new NameComparator());

        display(list);

    }

}



3)Write a Java program to find the first non-repeated character in a string using a HashMap.

String input = "aabbccddeffg";

Expected output = 'e';

import java.util.HashMap;

import java.util.Map;

public class NonRepChar {

    public static char firstNonRepeatedChar(String str) {

        Map<Character, Integer> charCountMap = new HashMap<>();

        // HashMap with character counts

        for (char c : str.toCharArray()) {

            charCountMap.put(c, charCountMap.getOrDefault(c, 0) + 1);

        }

        // non-repeated character

        for (char c : str.toCharArray()) {

            if (charCountMap.get(c) == 1) {

                return c;

            }

        }

        // Return a default value if no non-repeated character is found

        return '\0';

    }

    public static void main(String[] args) {

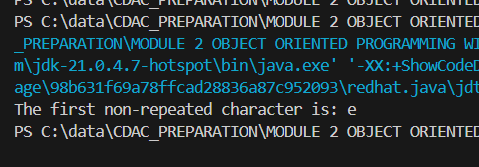
        String input = "aabbccddeffg";

        char result = firstNonRepeatedChar(input);

        System.out.println("The first non-repeated character is: " + result);

    }

}



4) Write a Java program that merges two sorted lists of integers into a single sorted list.

import java.util.Collection;

import java.util.List;

import java.util.ArrayList;

import java.util.Collections;

public class SortedList {

    public static void display(Collection<?> list) {

        for(Object element : list)

            System.out.print(element + "  ");

    }

    public static void main(String[] args) {

        List<Integer> list1 = new ArrayList<>();

        list1.add(50);

        list1.add(55);

        list1.add(60);

        list1.add(65);

        list1.add(70);

        List<Integer> list2 = new ArrayList<>();

        list2.add(55);

        list2.add(65);

        list2.add(75);

        list2.add(85);

        list2.add(95);

        System.out.println("Sorted List1");

        Collections.sort(list1);

        display(list1);

        System.out.println("\nSorted List 2");

        Collections.sort(list2);

        display(list2);

        list1.addAll(list2);

        System.out.println("\nSorted Combined List Elements :");

        Collections.sort(list1);

        display(list1);

    }

}

