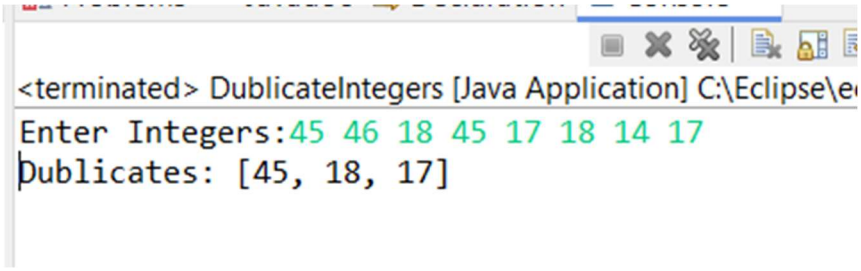
PG-DAC AUGUST 24 BATCH

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

Program code:

```
package org.collection.question1;

import java.util.ArrayList;
import java.util.HashSet;
import java.util.List;
import java.util.Scanner;
import java.util.Set;

public class DublicateIntegers {
	public static List<Integer>findDublictes(List<Integer> numbers){
		Set<Integer> S = new HashSet<>();
		List<Integer> dublicates = new ArrayList<>();

		for(Integer num : numbers) {
			if(!S.add(num)) {
				dublicates.add(num);
			}

		}

		return dublicates;
	}

	public static void main(String[] args) {
		Scanner sc = new Scanner(System.in);
		System.out.println("Enter Integers:");
		String[] input = sc.nextLine().split(" ");
		List<Integer> numbers = new ArrayList<>();
		for(String s : input) {
			numbers.add(Integer.parseInt(s));
		}

		List<Integer>dublicates = findDublictes(numbers);
		System.out.println("Dublicates: "  + dublicates);

		sc.close();

	}

}
```

Output:



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.

Program code:

```
package org.collection.question2;

import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;
import java.util.List;

public class Person {
   private String name;
   private int age;

   public Person(String name, int age) {
      this.name = name;
      this.age = age;
   }
   public String getName() {
      return name;
   }

   public int getAge() {
      return age;
   }

   @Override
   public String toString() {
      return name + " " + age ;
   }

   static class AgeComparator implements Comparator<Person> {
      public int compare(Person p1, Person p2) {

         int ageComparison = p1.getAge() - p2.getAge();

         if (ageComparison == 0) {
```

```
                return p1.getName().compareTo(p2.getName());
            }

            return ageComparison;
        }
    }

    public static void main(String[] args) {
        List<Person> per = new ArrayList<>();
        per.add(new Person("Rahul", 24));
        per.add(new Person("Siddh", 20));
        per.add(new Person("Rohit", 24));
        per.add(new Person("Balu", 22));

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

        System.out.println("Sorted list by age, then name: ");

        for (Person person : per) {
            System.out.println(person.getName() + " " + person.getAge());
        }
    }
}
```

Output:

```
<terminated> Person (1) [Java Application] C
Sorted list by age, then name:
Siddh 20
Balu 22
Rahul 24
Rohit 24
```

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

String input = "aabbccddeffg";
Expected output = 'e';

Program code:

```
package org.collection.question3;

import java.util.HashMap;

public class FirstNonRepeatedCharacter {
```

```
    public static char findFirstNonRepeatedChar(String input) {
        HashMap<Character, Integer> charCount = new HashMap<>();

        for (char ch : input.toCharArray()) {
            charCount.put(ch, charCount.getOrDefault(ch, 0) + 1);
        }

        for (char ch : input.toCharArray()) {
            if (charCount.get(ch) == 1) {
                return ch;
            }
        }

        return '0';
    }
    public static void main(String[] args) {
        String input = "aabbccddeffg";
        char result = findFirstNonRepeatedChar(input);

        if (result != '0') {
            System.out.println("First non-repeated character: " + result);
        } else {
            System.out.println("No non-repeated character found.");
        }
    }
}
```

Output:

```
<terminated> FirstNonRepeatedCharacter [Java
First non-repeated character: e
```

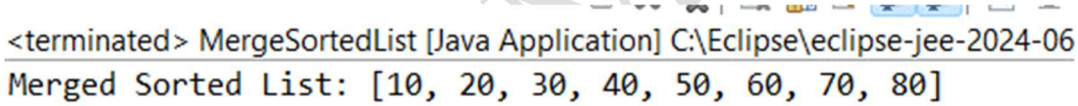
4) Write a Java program that merges two sorted lists of integers into a single sorted list.

Program code:

```
package org.collection.Question4;

import java.util.ArrayList;
import java.util.Collections;
import java.util.List;

public class MergeSortedList {
        public static List<Integer> mergeSortedList(List<Integer> list1, List<Integer>list2){

                List<Integer> mergedlist = new ArrayList<>(list1);
                mergedlist.addAll(list2);
                Collections.sort(mergedlist);
```

```
            return mergedlist;

        }

    public static void main(String[] args) {
        List<Integer> list1 = new ArrayList<>();
        list1.add(10);
        list1.add(30);
        list1.add(50);
        list1.add(70);

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

        list2.add(20);
        list2.add(40);
        list2.add(60);
        list2.add(80);

        List<Integer> mergedList = mergeSortedList(list1, list2);

        System.out.println("Merged Sorted List: " + mergedList);
    }
}
```

Output:



s