Collections Programs

1. List Practice Programs

a) Add Elements and Retrieve by Index

```
import java.util.ArrayList;
import java.util.List;

public class ListExample {
    public static void main(String[] args) {
        List<String> names = new ArrayList<>();
        names.add("Alice");
        names.add("Bob");
        names.add("Charlie");

        // Retrieve elements by index
        System.out.println("Element at index 1: " + names.get
(1)); // Output: Bob

        // Print all elements
        System.out.println("All elements in the list:");
        for (String name : names) {
            System.out.println(name);
        }
    }
}
```

b) Remove Duplicate Elements from an ArrayList

```
import java.util.ArrayList;
import java.util.HashSet;
import java.util.List;
```

```
import java.util.Set;

public class RemoveDuplicates {
    public static void main(String[] args) {
        List<String> list = new ArrayList<>();
        list.add("Java");
        list.add("Python");
        list.add("Java"); // Duplicate

        Set<String> uniqueElements = new HashSet<>(list);
        list.clear();
        list.addAll(uniqueElements);

        System.out.println("List after removing duplicates: "
+ list);
    }
}
```

c) Sort a List of Numbers

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

public class SortList {
    public static void main(String[] args) {
        List<Integer> numbers = new ArrayList<>();
        numbers.add(4);
        numbers.add(2);
        numbers.add(7);
        numbers.add(1);

        Collections.sort(numbers);
        System.out.println("Sorted List: " + numbers);
```

```
    }
}
```

---

2. Set Practice Programs

a) Add Elements and Check Uniqueness

```
import java.util.HashSet;
import java.util.Set;

public class SetExample {
    public static void main(String[] args) {
        Set<String> set = new HashSet<>();
        set.add("Java");
        set.add("Python");
        set.add("Java"); // Duplicate ignored

        System.out.println("Elements in set: " + set); // Out
put may vary
    }
}
```

b) Convert List to Set

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

public class ListToSet {
    public static void main(String[] args) {
        List<String> list = new ArrayList<>();
        list.add("A");
        list.add("B");
```

```
        list.add("A"); // Duplicate

        Set<String> set = new HashSet<>(list);
        System.out.println("Set after conversion from list: "
+ set);
    }
}
```

3. Map Practice Programs

a) Store and Retrieve Key-Value Pairs

```
import java.util.HashMap;
import java.util.Map;

public class MapExample {
    public static void main(String[] args) {
        Map<String, Integer> map = new HashMap<>();
        map.put("Apple", 3);
        map.put("Banana", 2);
        map.put("Orange", 5);

        System.out.println("Value for key 'Apple': " + map.ge
t("Apple"));

        System.out.println("All key-value pairs in map:");
        for (Map.Entry<String, Integer> entry : map.entrySet
()) {
            System.out.println(entry.getKey() + " -> " + entr
y.getValue());
        }
    }
}
```

b) Count Frequency of Characters in a String

```
import java.util.HashMap;
import java.util.Map;

public class CharacterFrequency {
    public static void main(String[] args) {
        String input = "hello world";
        Map<Character, Integer> frequencyMap = new HashMap<>
();

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

        System.out.println("Character frequencies:");
        for (Map.Entry<Character, Integer> entry : frequencyM
ap.entrySet()) {
            System.out.println(entry.getKey() + " -> " + entr
y.getValue());
        }
    }
}
```

---

4. Advanced Collection Programs

a) Convert Collection to Array

```
import java.util.ArrayList;
import java.util.List;

public class CollectionToArray {
    public static void main(String[] args) {
        List<Integer> list = new ArrayList<>();
```

```
        list.add(10);
        list.add(20);
        list.add(30);

        Integer[] array = list.toArray(new Integer[0]);

        System.out.println("Array elements:");
        for (int i : array) {
            System.out.println(i);
        }
    }
}
```

b) Union of Two Sets

```
import java.util.HashSet;
import java.util.Set;

public class SetUnion {
    public static void main(String[] args) {
        Set<Integer> set1 = new HashSet<>();
        set1.add(1);
        set1.add(2);
        set1.add(3);

        Set<Integer> set2 = new HashSet<>();
        set2.add(3);
        set2.add(4);
        set2.add(5);

        Set<Integer> unionSet = new HashSet<>(set1);
        unionSet.addAll(set2);

        System.out.println("Union of set1 and set2: " + union
Set);
```

```
    }
}
```

c) Intersection of Two Sets

```
import java.util.HashSet;
import java.util.Set;

public class SetIntersection {
    public static void main(String[] args) {
        Set<Integer> set1 = new HashSet<>();
        set1.add(1);
        set1.add(2);
        set1.add(3);

        Set<Integer> set2 = new HashSet<>();
        set2.add(3);
        set2.add(4);
        set2.add(5);

        Set<Integer> intersectionSet = new HashSet<>(set1);
        intersectionSet.retainAll(set2);

        System.out.println("Intersection of set1 and set2: "
+ intersectionSet);
    }
}
```

d) Difference of Two Sets

```
import java.util.HashSet;
import java.util.Set;

public class SetDifference {
    public static void main(String[] args) {
```

```
        Set<Integer> set1 = new HashSet<>();
        set1.add(1);
        set1.add(2);
        set1.add(3);

        Set<Integer> set2 = new HashSet<>();
        set2.add(3);
        set2.add(4);
        set2.add(5);

        Set<Integer> differenceSet = new HashSet<>(set1);
        differenceSet.removeAll(set2);

        System.out.println("Difference of set1 and set2: " +
differenceSet);
    }
}
```

These programs should give you solid practice with Java collections, covering each major class and operation. Let me know if you'd like more examples or any additional details!