Practical 9:- Collection API

1.Write a program to demostrate user of ArrayList, LinkedList ,LinkedHashMap, TreeMap and HashSet Class. And also implement CRUD operation without database connection using Collection API.

```
import java.util.ArrayList;
import java.util.LinkedList;
import java.util.LinkedHashMap;
import java.util.TreeMap; import
java.util.HashSet; class
CollectionDemo {
   public static void main(String[] args) {

      // ArrayList
      ArrayList<String> arrayList = new ArrayList<String>();
arrayList.add("Apple");       arrayList.add("Banana");
arrayList.add("Orange");
      System.out.println("ArrayList: " + arrayList);

      // LinkedList
      LinkedList<Integer> linkedList = new LinkedList<Integer>();
linkedList.add(1);       linkedList.add(2);       linkedList.add(3);
      System.out.println("LinkedList: " + linkedList);
      // LinkedHashMap
      LinkedHashMap<String, Integer> linkedHashMap = new LinkedHashMap<String,
Integer>();
      linkedHashMap.put("aman", 17);
linkedHashMap.put("maiz", 26);
linkedHashMap.put("faizan", 25);
      System.out.println("LinkedHashMap: " + linkedHashMap);
      // TreeMap
      TreeMap<Integer, String> treeMap = new TreeMap<Integer, String>();
treeMap.put(1, "One");       treeMap.put(3, "Three");       treeMap.put(2,
"Two");
      System.out.println("TreeMap: " + treeMap);

      // HashSet
      HashSet<String> hashSet = new HashSet<String>();
hashSet.add("Black");       hashSet.add("White");
hashSet.add("Blue");
      System.out.println("HashSet: " + hashSet);

      // CRUD operations
      System.out.println("\nCRUD operations using Collection API:\n");
```

```
  // Create
      ArrayList<String> fruits = new ArrayList<String>();
fruits.add("Apple");       fruits.add("Banana");
fruits.add("Orange");
      System.out.println("Created ArrayList: " + fruits);
      // Read
      String fruit = fruits.get(1);
      System.out.println("Read element at index 1: " + fruit);
      // Update
      fruits.set(1, "Grape");
      System.out.println("Updated ArrayList: " + fruits);
      // Delete
fruits.remove(2);
      System.out.println("Deleted element at index 2: " + fruits);
  } }
```

Output :-

```
D:\12302130501038\prac 6>javac CollectionDemo.java

D:\12302130501038\prac 6>java CollectionDemo
ArrayList: [Apple, Banana, Orange]
LinkedList: [1, 2, 3]
LinkedHashMap: {aman=17, maiz=26, faizan=25}
TreeMap: {1=One, 2=Two, 3=Three}
HashSet: [White, Blue, Black]

CRUD operations using Collection API:

Created ArrayList: [Apple, Banana, Orange]
Read element at index 1: Banana
Updated ArrayList: [Apple, Grape, Orange]
Deleted element at index 2: [Apple, Grape]

D:\12302130501038\prac 6>
```

2.Write a program to Sort Array,ArrayList,String,List,Map and Set

```
import java.util.*;

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

      // Sort an array of integers
int[] intArray = { 85, 25, 83, 17, 9 };
      Arrays.sort(intArray);
      System.out.println("Sorted Array of Integers: " + Arrays.toString(intArray));

      // Sort an ArrayList of strings
      List<String> stringList = new ArrayList<>(Arrays.asList("dog", "cat", "zebra", "lion"));
      Collections.sort(stringList);
      System.out.println("Sorted ArrayList of Strings: " + stringList);

      // Sort an array of strings
      String[] stringArray = { "dog", "cat", "zebra", "lion" };
      Arrays.sort(stringArray);
      System.out.println("Sorted Array of Strings: " + Arrays.toString(stringArray));

      // Sort a List of integers
      List<Integer> integerList = new ArrayList<>(Arrays.asList(5, 2, 8, 1, 9));
      Collections.sort(integerList);
      System.out.println("Sorted List of Integers: " + integerList);

      // Sort a Map by its keys
      Map<String, Integer> integerMap = new HashMap<>();
integerMap.put("dog", 25);       integerMap.put("cat", 9);
integerMap.put("zebra", 83);
      integerMap.put("lion", 17);
      TreeMap<String, Integer> sortedMap = new TreeMap<>(integerMap);
      System.out.println("Sorted Map by Keys: " + sortedMap);

      // Sort a Set of strings
      Set<String> stringSet = new HashSet<>(Arrays.asList("dog", "cat", "zebra", "lion"));
      TreeSet<String> sortedSet = new TreeSet<>(stringSet);
      System.out.println("Sorted Set of Strings: " + sortedSet);
   }
}
```

Output :-

```
D:\12302130501038\prac 6>javac SortExample.java

D:\12302130501038\prac 6>java SortExample
Sorted Array of Integers: [9, 17, 25, 83, 85]
Sorted ArrayList of Strings: [cat, dog, lion, zebra]
Sorted Array of Strings: [cat, dog, lion, zebra]
Sorted List of Integers: [1, 2, 5, 8, 9]
Sorted Map by Keys: {cat=9, dog=25, lion=17, zebra=83}
Sorted Set of Strings: [cat, dog, lion, zebra]

D:\12302130501038\prac 6>
```