ASSIGNMENT-14

20BCSE50_Kumar Jijnasu_CSE_C1-08

1..

```
import java.io.*;
import java.util.*;
class DemoArrayList {

    public static void main(String[] args) {

        try {

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

            list.add(1);
            list.add(2);
            list.add(3);
            list.add(4);

            System.out.println(list);
            list.set(2, 200);

            System.out.println(list);

        }

        catch (Exception e) {

        System.out.println(e);
        }
    }
}
```

2..

```
import java.util.*;
  public class Collection2 {
  public static void main(String[] args) {
   LinkedList<String> c1= new LinkedList<String>();
          c1.add("Red");
          c1.add("Green");
          c1.add("Black");
          c1.add("White");
          c1.add("Pink");
          LinkedList<String> c2= new LinkedList<String>();
          c2.add("Red");
          c2.add("Green");
          c2.add("Black");
          c2.add("Orange");
         LinkedList<String> c3= new LinkedList<String>();
         c3.add("Red");
         c3.add("Green");
         c3.add("Black");
         c3.add("Pink");
         c3.add("Black");
         c3.add("White");


          System.out.println(compare(c1,c2));
          System.out.println(compare(c1,c3));
     }


     static boolean compare(LinkedList<String> c1,LinkedList<String> c2)
     {
         boolean res = true;
         for (String e : c1)
         {
            res = (c2.contains(e) ? true : false);
            if(!res)
               break;
         }
         if(res)
            for (String e : c2)
            {
               res = (c1.contains(e) ? true : false);
               if(!res)
                  break;
            }
         return res;
     }
}
```

3..

```
import java.util.TreeSet;
import java.util.Iterator;

  public class Collection3 {
  public static void main(String[] args) {
 // creating TreeSet
   TreeSet <Integer>tree_num = new TreeSet<Integer>();
   // Add numbers in the tree
   tree_num.add(10);
   tree_num.add(22);
   tree_num.add(36);
   tree_num.add(25);
   tree_num.add(16);
   tree_num.add(70);
   tree_num.add(82);
   tree_num.add(89);
   tree_num.add(14);
   System.out.println("Original tree set: "+tree_num);
   System.out.println("Removes the first(lowest) element: "+tree_num.pollFirst());
   System.out.println("Tree set after removing first element: "+tree_num);
   }
}
```

4..

```
import java.util.*;
public class Collection4
{
  public static void main(String[] args)
    {
        String s = "Demo for stack in java";
        reverse(s);
    }
    public static void reverse(String s)
    {
    Stack<String> stc = new Stack<>();
    String temp = "";
    for(int i = 0; i < s.length(); i++)
    {
        if(s.charAt(i) == ' ')
        {
        stc.add(temp);

        temp = "";
        }
        else
        {
        temp = temp + s.charAt(i);
        }

    }

    stc.add(temp);

    while(!stc.isEmpty())
    {
        temp = stc.peek();
        System.out.print(temp + " ");
        stc.pop();
    }

    System.out.println();
    }

}
```

5..

```
import java.util.*;
public class Colllection5
{
 public static void main(String args[])
 {
  TreeMap<Integer,String> treemap=new TreeMap<Integer,String>();
  treemap.put(1, "A");
  treemap.put(2, "B");
  treemap.put(3, "C");
  treemap.put(4, "C");
  treemap.put(5, "D");

  System.out.println("Elements of TreeMap content are: " + treemap);
  System.out.println("Keys which are between 2 and 4(inclusive): " +
treemap.tailMap(2).headMap(4+1));
  }
}
```

6..

```
import java.util.*;
public class Collection6{
    public static void main(String args[]){
        Set<String> h1=new HashSet<String>();
        h1.add("Red");
        h1.add("Green");
        h1.add("Black");
        System.out.println( "first hashset is: " + h1);
        Set<String> h2=new HashSet<String>();
        h2.add("Blue");
        h2.add("Red");
        h2.add("Black");
        System.out.println( "second hashset is: " + h2);
        h1.retainAll(h2);
        Set<String> hashSetToTreeSet= new TreeSet<>(h1);
        System.out.println("tree set: " + hashSetToTreeSet);
    }
}
```