Problem Statement

1. Write a simple Timer that can periodically print a timeout message.

2. Write a program to build any collection containing duplicates. Create its copy with all duplicates Removed

Solution

1. Write a simple Timer that can periodically print a timeout message.

```
public class SimpleTimer
{
 public static void main(String[] args) //main class
 {
  Thread t1 = new Thread(new Runnable()
  {
@Override// this will override the functions
  public void run() {
  for(int i = 0; i < 5; i++)
  {
  try {
  Thread.sleep(30000);
    }
catch (InterruptedException e)
{
  e.printStackTrace();
   }
 System.out.println("Simple Timer that can periodically print a timeout message T1 : "+i);
  }
    }
  });// using runnable method which will use thread function
```

```
Thread t2 = new Thread(new Runnable() {
@Override
  public void run() {
   for(int i = 0; i < 5; i++) {
    try {
    Thread.sleep(30000);
    }
catch (InterruptedException e)
{
 e.printStackTrace();
             }
   System.out.println("Simple Timer that can periodically print a timeout message T2 : "+i);
          }
        }
      });
  t1.start();//calling fuction
   t2.start();//calling fuction
System.out.println("The statement will print in after every 30 seconds cycle");
   }
}
```

```
C:\sweety_backup\Training\Assignment5>javac SimpleTimer.java

C:\sweety_backup\Training\Assignment5>java SimpleTimer
The statement will print in after every 30 seconds cycle
Simple Timer that can periodically print a timeout message T1 : 0
Simple Timer that can periodically print a timeout message T2 : 0
Simple Timer that can periodically print a timeout message T1 : 1
Simple Timer that can periodically print a timeout message T2 : 1
Simple Timer that can periodically print a timeout message T1 : 2
Simple Timer that can periodically print a timeout message T2 : 2
```

2.Write a program to build any collection containing duplicates. Create its copy with all duplicates Removed.

```
import java.util.ArrayList;
import java.util.Arrays;
import java.util.LinkedHashSet;
import java.util.List;
public class RemoveDuplicatesFromArrayList
 {
 public static void main(String args[]) {
List<String> duplicateList = (List<String>) Arrays.asList("Anish" , "Anita", "Anokhi", "shreya","Monty","Anish");//ArrayList with duplicates String
System.out.println("size of Arraylist with duplicates: " + duplicateList.size());      //print size of Arraylist
System.out.println("ArrayList with duplicates: " + duplicateList);//print data in arraylist
 LinkedHashSet<String> listToSet = new LinkedHashSet<String>(duplicateList);// //Converting ArrayList to HashSet to remove duplicates as hashset dont keep duplicate
 List<String> listWithoutDuplicates = new ArrayList<String>(listToSet);
   System.out.println("size of ArrayList without duplicates: " + listToSet.size());
      System.out.println("ArrayList after removing duplicates in same order: " + listWithoutDuplicates);
   }
}
```

Screenshot for the Output

```
C:\sweety_backup\Training\Assignment5>javac RemoveDuplicatesFromArrayList.java

C:\sweety_backup\Training\Assignment5>java RemoveDuplicatesFromArrayList
size of Arraylist with duplicates: 9
ArrayList with duplicates: [Anish, Anita, Anokhi, shreya, Monty, Anish, Aman, Sw
eety, Shreya]
size of ArrayList without duplicates: 8
ArrayList after removing duplicates in same order: [Anish, Anita, Anokhi, shreya
, Monty, Aman, Sweety, Shreya]
```