

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

1.1 Source Code: file name – MyTimer.java

```
//Objective is to print Timeout Message periodically
import java.util.Date;
import java.util.Timer;
import java.util.TimerTask;
public class MyTimer extends TimerTask //TimerTask is a Java Class for
scheduling or running a particular task
{

    @Override
    public void run()
    {
        System.out.println("Timer task started at:"+new Date());
        completeTask();
        System.out.println("Timeout at:"+new Date());
    }
    private void completeTask()
    {
        try
        {
            Thread.sleep(10000); // for every 10 seconds, we need to print
timeout message
            // this may throw InterruptedException if any thread interrupts
the timer.
        }
        catch (InterruptedException e)
        {
            e.printStackTrace();
        }
    }
    public static void main(String args[])
    {
        MyTimer timerTask = new MyTimer();
        //running timer task as daemon thread
        Timer timer = new Timer(true);
        timer.scheduleAtFixedRate(timerTask, 0, 10*1000);
        System.out.println("*** TimerTask started for 1 Minute ***");
        //cancel after sometime
        try {
            Thread.sleep(60000); //this is used to stop the Timer after 1min -
60000 millisecs
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
        timer.cancel();
        System.out.println("*** TimerTask completed after 1 Minute ***");
    }
}
```

1.2 Output:

```
*** TimerTask started for 1 Minute ***
Timer task started at:Sun Nov 25 14:49:21 IST 2018
```

```

Timeout at:Sun Nov 25 14:49:31 IST 2018
Timer task started at:Sun Nov 25 14:49:31 IST 2018
Timeout at:Sun Nov 25 14:49:41 IST 2018
Timer task started at:Sun Nov 25 14:49:41 IST 2018
Timeout at:Sun Nov 25 14:49:51 IST 2018
Timer task started at:Sun Nov 25 14:49:51 IST 2018
Timeout at:Sun Nov 25 14:50:01 IST 2018
Timer task started at:Sun Nov 25 14:50:01 IST 2018
Timeout at:Sun Nov 25 14:50:11 IST 2018
Timer task started at:Sun Nov 25 14:50:11 IST 2018
*** TimerTask completed after 1 Minute ***

```

1.3 Output Screenshots:

The screenshot displays an IDE with the 'MyTimer.java' file open. The code defines a class 'MyTimer' that extends 'TimerTask'. It includes an '@Override' annotation for the 'run()' method, which prints 'Timer task started at:' followed by the current date and calls 'completeTask()'. The 'completeTask()' method is private and contains a try-catch block. The try block calls 'Thread.sleep(10000);' with a comment explaining the 10-second interval for printing timeout messages. The catch block handles 'InterruptedException' by printing the stack trace. The console output at the bottom shows the execution of the program, starting with '*** TimerTask started for 1 Minute ***', followed by alternating 'Timer task started at:' and 'Timeout at:' messages every 10 seconds, and ending with '*** TimerTask completed after 1 Minute ***'.

```

Even.java Shape.java AgeException.java Collection.java MyTimer.java x
1 Objective is to print Timeout Message periodically
2 port java.util.Date;
3 port java.util.Timer;
4 port java.util.TimerTask;
5 blic class MyTimer extends TimerTask //TimerTask is a Java Class for scheduling or running
6
7
8 @Override
9 public void run()
10 {
11     System.out.println("Timer task started at:"+new Date());
12     completeTask();
13     System.out.println("Timeout at:"+new Date());
14 }
15 private void completeTask()
16 {
17     try
18     {
19         Thread.sleep(10000); // for every 10 seconds, we need to print timeout message
20         // this may throw InterruptedException if any thread interrupts the timer.
21     }
22     catch (InterruptedException e)
23     {
24         e.printStackTrace();
25     }
26 }

```

Problems @ Javadoc Declaration Console x

```

<terminated> MyTimer [Java Application] C:\Program Files\Java\jdk-11.0.1\bin\javaw.exe (Nov 25, 2018, 2:49:20 PM)
*** TimerTask started for 1 Minute ***
Timer task started at:Sun Nov 25 14:49:21 IST 2018
Timeout at:Sun Nov 25 14:49:31 IST 2018
Timer task started at:Sun Nov 25 14:49:31 IST 2018
Timeout at:Sun Nov 25 14:49:41 IST 2018
Timer task started at:Sun Nov 25 14:49:41 IST 2018
Timeout at:Sun Nov 25 14:49:51 IST 2018
Timer task started at:Sun Nov 25 14:49:51 IST 2018
Timeout at:Sun Nov 25 14:50:01 IST 2018
Timer task started at:Sun Nov 25 14:50:01 IST 2018
Timeout at:Sun Nov 25 14:50:11 IST 2018
Timer task started at:Sun Nov 25 14:50:11 IST 2018
*** TimerTask completed after 1 Minute ***

```

2. Problem Statement: Write a program to build any collection containing duplicates. Create its copy with all duplicates removed

2.1 Source Code: file name – Collection.java

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

public class Collection{

    public static void main(String args[]){

        // Declaring a type of collection - list - ArrayList with String datatype
        System.out.println("*** Collection with duplicates ***");
        List<String> names = new ArrayList<String>();
        //Adding values to ArrayList
        names.add("Subbarao");
        names.add("Harisha");
        names.add("Vedansh");
        names.add("Rishika");
        names.add("Jahnavi");
        names.add("Vedansh");
        names.add("Anusha");
        names.add("Jyothi");
        names.add("Rishika");
        names.add("Jahnavi");
        List<Integer> num=new ArrayList<Integer>();
        num.add(2);
        num.add(4);
        num.add(2);
        num.add(6);
        num.add(3);
        num.add(8);
        num.add(5);
        num.add(5);
        // Printing the ArrayList
        System.out.println("List of the Names : " +names);
        System.out.println("List of the Numbers : " +num);
        //objective is to remove the duplicates from the collection.
        // let us use LinkedHashSet for this. As this set will not allow to
        insert the duplicates
        // and also the order of the list provided will be same.
        System.out.println("*** Collection without duplicates ***");
        Set<String> namesWithoutDup = new LinkedHashSet<String>(names);
        //copied the arraylist into LinkedHashSet
        names.clear();// clear the arraylist
        names.addAll(namesWithoutDup); //copying the values without duplicates
        into ArrayList
        Set<Integer> numWithoutDuplicates=new LinkedHashSet<Integer>(num);
        num.clear();
        num.addAll(numWithoutDuplicates);
        System.out.println("list of Names without duplicates :");
```

```

        for(int i=0;i<names.size();i++)
        System.out.println(names.get(i)+"");
        System.out.println("list of Numberes without duplicates :");
        for(int i=0;i<num.size();i++)
        System.out.println(num.get(i)+"");
    }
}

```

2.2 Output:

```

*** Collection with duplicates ***
List of the Names : [Subbarao, Harisha, Vedansh, Rishika, Jahnavi, Vedansh,
Anusha, Jyothi, Rishika, Jahnavi]
List of the Numbers : [2, 4, 2, 6, 3, 8, 5, 5]
*** Collection without duplicates ***
list of Names without duplicates :
Subbarao
Harisha
Vedansh
Rishika
Jahnavi
Anusha
Jyothi
list of Numberes without duplicates :
2
4
6
3
8
5

```

2.3 Output Screenshots:

```
Even.java Shape.java AgeException.java Collection.java
1 import java.util.ArrayList;
2 import java.util.LinkedHashSet;
3 import java.util.List;
4 import java.util.Set;
5
6 public class Collection
7 {
8     public static void main(String args[])
9     {
10         // Declaring a type of collection - list - ArrayList with String datatype
11         System.out.println("*** Collection with duplicates ***");
12         List<String> names = new ArrayList<String>();
13         //Adding values to ArrayList
14         names.add("Subbarao");
15         names.add("Harisha");
16         names.add("Vedansh");
17         names.add("Rishika");
18         names.add("Jahnavi");
19         names.add("Vedansh");
20     }
21 }
```

Problems @ Javadoc Declaration Console

<terminated> Collection [Java Application] C:\Program Files\Java\jdk-11.0.1\bin\javaw.exe (Nov 25, 2018, 11:59:54 AM)

*** Collection with duplicates ***

List of the Names : [Subbarao, Harisha, Vedansh, Rishika, Jahnavi, Vedansh, Anusha, Jyothi, Rishika, Jahnavi]

List of the Numbers : [2, 4, 2, 6, 3, 8, 5, 5]

*** Collection without duplicates ***

list of Names without duplicates :

Subbarao

Harisha

Vedansh

Rishika

Jahnavi

Anusha

Jyothi

list of Numberes without duplicates :

2

4

6

3

8

5

****END OF DOCUMENT****