EX NO:8	MULTITHREADED APPLICATION
DATE: 16-09-19	

AIM:

To develop a java program that implements a multi-threaded application that has three threads.

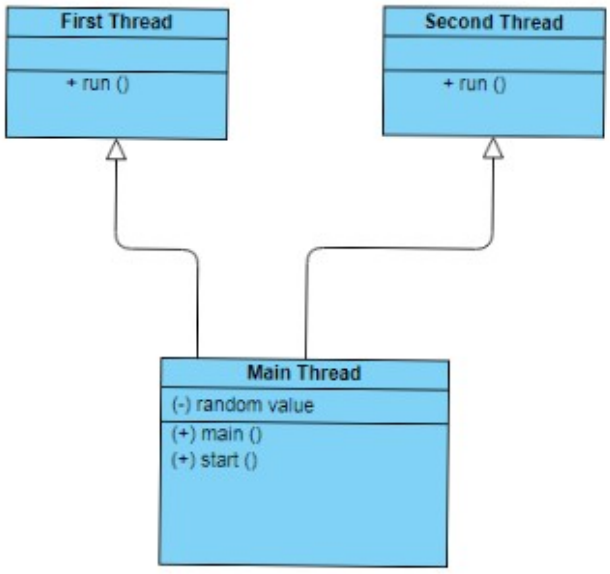
REQUIREMENT:

Develop a java program that implements a multithread application that has 3 threads.First generates a random integer for every 1 second and if the value is even,second thread computes the square of the number and prints.If the value is odd,the third thread will print the value of cube of the number.

ALGORITHM:

STEP 1: Declare a package called multithread.
STEP 2: Declare a class name of FirstThread,SecondThread which extends from Thread.
STEP 3: Declare a object in the respective classes.
STEP 4: Create a condition to check the statements.
STEP 5: Print the result.

CLASSDIAGRAM:



PROGRAM:

```
/***
* EXPERIMENT-08
*developed by Nithishkumar
*Saveetha Engineering College
*jpnithishkumar@gmail.com
*/
package multithread;

import java.util.*;

public class MainThread {

     public static Integer RandomValue;

     public static void main(String[] args) {
          EvenThread t1;
          OddThread t2;
          Random r;

               t1=new EvenThread();
               t2=new OddThread();

               r=new Random();
               RandomValue=-1;

               t1.start();
               t2.start();

               try
               {
               while(true)
               {
                    synchronized(RandomValue)
                    {
                         if(RandomValue==-1)
                         {
                              RandomValue=r.nextInt(10);
                              System.out.println("Placed a new
number "+RandomValue);
                         }
                    }
                    Thread.sleep(4000);
               }
               }catch(InterruptedException ex)
               {
                    System.out.println("Error:"+ex);
               }

     }
```

```
}



/***
* EXPERIMENT-08
*developed by Nithishkumar
*Saveetha Engineering College
*jpnithishkumar@gmail.com
*/
package multithread;

public class OddThread extends Thread {

     public void run()
     {
          try
          {
               System.out.println("Second thread started...");
               while(true)
               {
                    synchronized(MainThread.RandomValue)
                    {
                         if(MainThread.RandomValue%2!
=0&&MainThread.RandomValue!=-1)
                         {
                              System.out.println("Value is Odd
Number");
                              System.out.println("Answer="+
(MainThread.RandomValue*MainThread.RandomValue*MainThread.RandomVa
lue));
                              MainThread.RandomValue=-1;
                         }
                    }

                    Thread.sleep(2000);
               }
          }catch(InterruptedException ex)
          {
               System.out.println("Error:"+ex);
          }
     }
}


/***
* EXPERIMENT-08
*developed by Nithishkumar
*Saveetha Engineering College
*jpnithishkumar@gmail.com
*/
```

```
package multithread;

public class EvenThread extends Thread {

     public void run()
     {

          try
          {
               System.out.println("First thread started...");
               while(true)
               {
                    synchronized(MainThread.RandomValue)
                    {
                         if(MainThread.RandomValue
%2==0&&MainThread.RandomValue!=-1)
                         {
                              System.out.println("Value is Even
Number");
                              System.out.println("Answer="+
(MainThread.RandomValue*MainThread.RandomValue));
                              MainThread.RandomValue=-1;
                         }
                    }

                    Thread.sleep(2000);
               }
          }catch(InterruptedException ex)
          {
               System.out.println("Error:"+ex);
          }

     }
}
```

OUTPUT:

```
Placed a new number 5
Second thread started...
Value is Odd Number
First thread started...
Answer=125
Placed a new number 8
Value is Even Number
Answer=64
Placed a new number 3
Value is Odd Number
Answer=27
Placed a new number 8
Value is Even Number
Answer=64
```

RESULT:

Thus a java application that performs multithreading is developed.