EX NO: 08
DATE : 12-09-19

MULTI THREADED APPLICATION

Aim:

To develop a java application that implements a multi threaded application that has three threads. First thread generates a random integer every one 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 as multithread.

Step 2: Declare the class as super class and subclass.

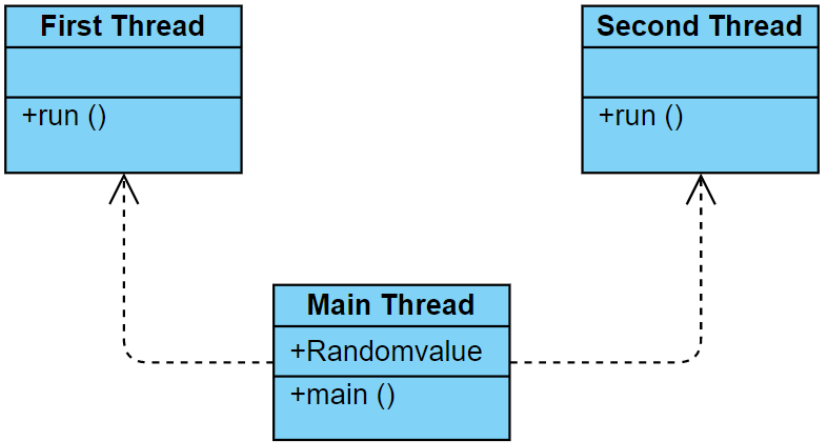
Step 3: Declare a constructor and add the data members.

Step 4: Inherit the class from the superclass and add the data members.

Step 5: Calculate the required multithread application.

Step 6: Display the result.

CLASS DIAGRAM:



PROGRAM:

```
/*
 * Program to perform first thread
 *  By manikanta
```

```
 *  amanikanta69@gmail.com
 */
package multithread;

public class FirstThread 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");
                              System.out.println("Answer="+
(MainThread.RandomValue*MainThread.RandomValue));
                              MainThread.RandomValue=-1;
                         }
                    }

                    Thread.sleep(1000);
               }
          }catch(InterruptedException ex)
          {
               System.out.println("Error:"+ex);
          }

     }
}
/*
 * Program to perform main thread
 *  By manikanta
 *  amanikanta69@gmail.com
 */
package multithread;
```

```
import java.util.*;

public class MainThread {

	public static Integer RandomValue;

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

			t1=new FirstThread();
			t2=new SecondThread();

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

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

			try
			{
			while(true)
			{
				synchronized(RandomValue)
				{
					if(RandomValue==-1)
					{
						RandomValue=r.nextInt(200);
						System.out.println("Placed a new
number "+RandomValue);
					}
				}
				Thread.sleep(1000);
			}
			}catch(InterruptedException ex)
			{
				System.out.println("Error:"+ex);
			}

	}
```

```
}
/*
 * Program to perform second thread
 *  By manikanta
 *  amanikanta69@gmail.com
 */
package multithread;

public class SecondThread 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");
						System.out.println("Answer="+
(MainThread.RandomValue*MainThread.RandomValue*MainThread.Random
Value));
						MainThread.RandomValue=-1;
					}
				}

				Thread.sleep(1000);
			}
		}catch(InterruptedException ex)
		{
			System.out.println("Error:"+ex);
		}
	}
}
```

OUTPUT:

First thread started…

Second thread started…

Value is odd

Answer=857375

Placed a new number 95

Placed a new number 71

Value is odd

Answer=357911

Placed a new number 100

Value is even

Answer=10000

Placed a new number 191

Value is odd

Answer=6967871

Placed a new number 136

Value is even

Answer=18496

Placed a new number 29

RESULT:

Thus a java console application that implements a multi-threaded application is verified.