

Ex No: 8	MULTITHREADED APPLICATION
Date: 16/09/2019	

Aim:

*To develop a java program for implementing multithread application.

Requirements:

*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.

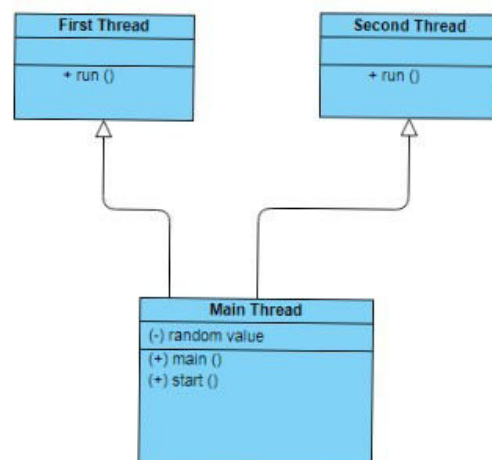
STEP 2: Declare a class name of FirstThread, SecondThread which extends from Thread.

STEP 3: Declare a object in the respective classes.

STEP 4: Create a condition to check the statements.

STEP 5: Print the result.

Class Diagram:



Program:

```
/**
 *Developed by
 *D. Sarathi Raj
 *212217105054
 *Saveetha Engineering College
 *sarathiraj852000@gmail.com
 */
package multithread;

import java.util.Random;

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);
        }

    }

}

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

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.RandomValue));
                        MainThread.RandomValue=-1;
                    }
                }

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

```

```
}  
}
```

Output:

```
First thread started...  
Second thread started...  
Placed a new number 43  
Value is odd  
Answer=79507  
Placed a new number 56  
Value is even  
Answer=3136  
Placed a new number 155  
Value is odd  
Answer=3723875  
Placed a new number 96  
Value is even  
Answer=9216  
Placed a new number 33  
Value is odd  
Answer=35937  
Placed a new number 174  
Value is even  
Answer=30276  
Placed a new number 157  
Value is odd  
Answer=3869893  
Placed a new number 70  
Value is even  
Answer=4900  
Placed a new number 164  
Value is even  
Answer=26896  
Placed a new number 141  
Value is odd  
Answer=2803221  
Placed a new number 101  
Value is odd  
Answer=1030301  
Placed a new number 8  
Value is even  
Answer=64  
Placed a new number 34  
Value is even  
Answer=1156  
Placed a new number 164  
Value is even  
Answer=26896  
Placed a new number 51  
Value is odd  
Answer=132651  
Placed a new number 139  
Value is odd  
Answer=2685619  
Placed a new number 110  
Value is even  
Answer=12100  
Placed a new number 134
```

Value is even
Answer=17956
Placed a new number 144
Value is even
Answer=20736
Placed a new number 16
Value is even
Answer=256
Placed a new number 190
Value is even
Answer=36100
Placed a new number 73
Value is odd
Answer=389017
Placed a new number 70
Value is even
Answer=4900
Placed a new number 44
Value is even
Answer=1936
Placed a new number 101
Value is odd
Answer=1030301
Placed a new number 48
Value is even
Answer=2304

Result:

*Thus, a java application that performs multithreading is written and executed successfully.