Ex.No: 9	**Java Application for Multithreading**
Date:	

Aim:

To create a Java console application the uses the multithreading concepts in java. The Application has 3 threads one creates random number, one thread computes square of that number and another one computes the cube of that number.

Algorithm:

Step 1 Start the Process

Step 2 Create a thread that generates random number

Step 3 Obtain one random number and check is odd or even

- Step 3.1 If number is even then create and start thread that computes square of a number
- Step 3.2 Compute number * number and display the answer
- Step 3.3 Notify to Random number thread and goto step 4
- Step 3.4 If number is odd then create and start thread that computes cube of a number
- Step 3.4 Compute number * number * number and display the answer
- Step 3.5 Notify to Random number thread and goto step 4

Step 4 Wait for 1 Second and Continue to Step 3 until user wants to exits.

Step 5 Stop the Process

Coding:

RandomNumberThread.java

```
package com.raja.oopslab.threading;

import java.util.Random;

public class RandomNumberThread extends Thread{
	Random num = new Random();
	int value;
	@Override
	public void run(){
		while(true){
			try {
				this.sleep(1000);
			} catch (InterruptedException e) {

			}
			value = num.nextInt(1000);
			System.out.println("RandomNumberThread generated a number "+value);
			if(value % 2 == 0)
				new SquareGenThread(value).start();
			else
				new CubeGenThread(value).start();
		}

	}
}
```

SquareGenThread.java

```
package com.raja.oopslab.threading;

public class SquareGenThread extends Thread{
	int number;
	int squre;
	public SquareGenThread(int number) {
		this.number = number;
	}
	@Override
	public void run(){
		try {
			this.sleep(3000);
		} catch (InterruptedException e) {

		}
		this.squre = this.number * this.number;
		System.out.println("SquareGenThread--> Square of "+number+" is "+squre);
	 }
}
```

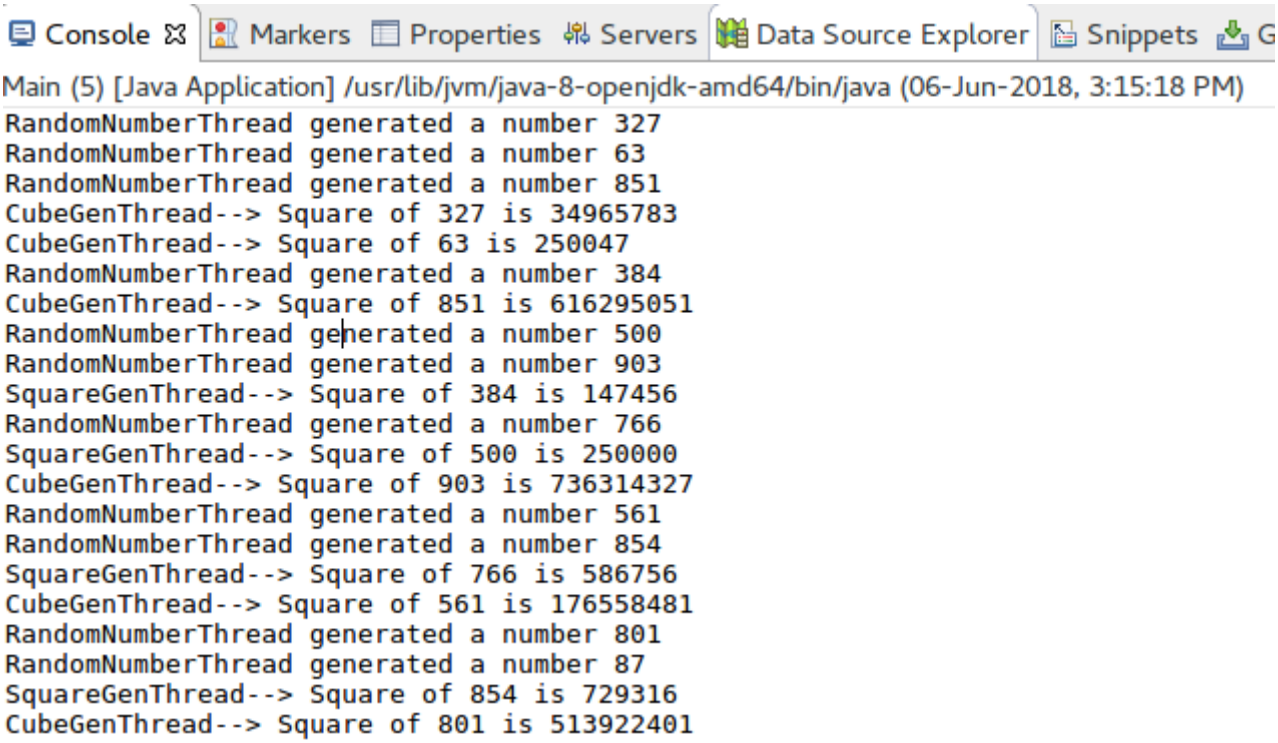
CubeGenThread.java

```
package com.raja.oopslab.threading;

public class CubeGenThread extends Thread{
	int number;
	int squre;
	public CubeGenThread(int number) {
		this.number = number;
	}
	@Override
	public void run(){
		try {
			this.sleep(2000);
		} catch (InterruptedException e) {

		}
		this.squre = this.number * this.number * this.number;
		System.out.println("CubeGenThread--> Square of "+number+" is "+squre);
	 }
}
```

Main.java

```
import java.util.Random;
import com.raja.oopslab.threading.RandomNumberThread;
public class Main {
	public static void main(String[] args) {
		new RandomNumberThread().start();
	}
}
```

Output:



Result:

The java console application for multithreading was developed and tested successfully.