DATE: 31-03-2023 **EXPERIMENT NO** 9 **REGISTER NO**: 2162014

IMPLEMENTATION OF MULTITHREADED PROGRAMS

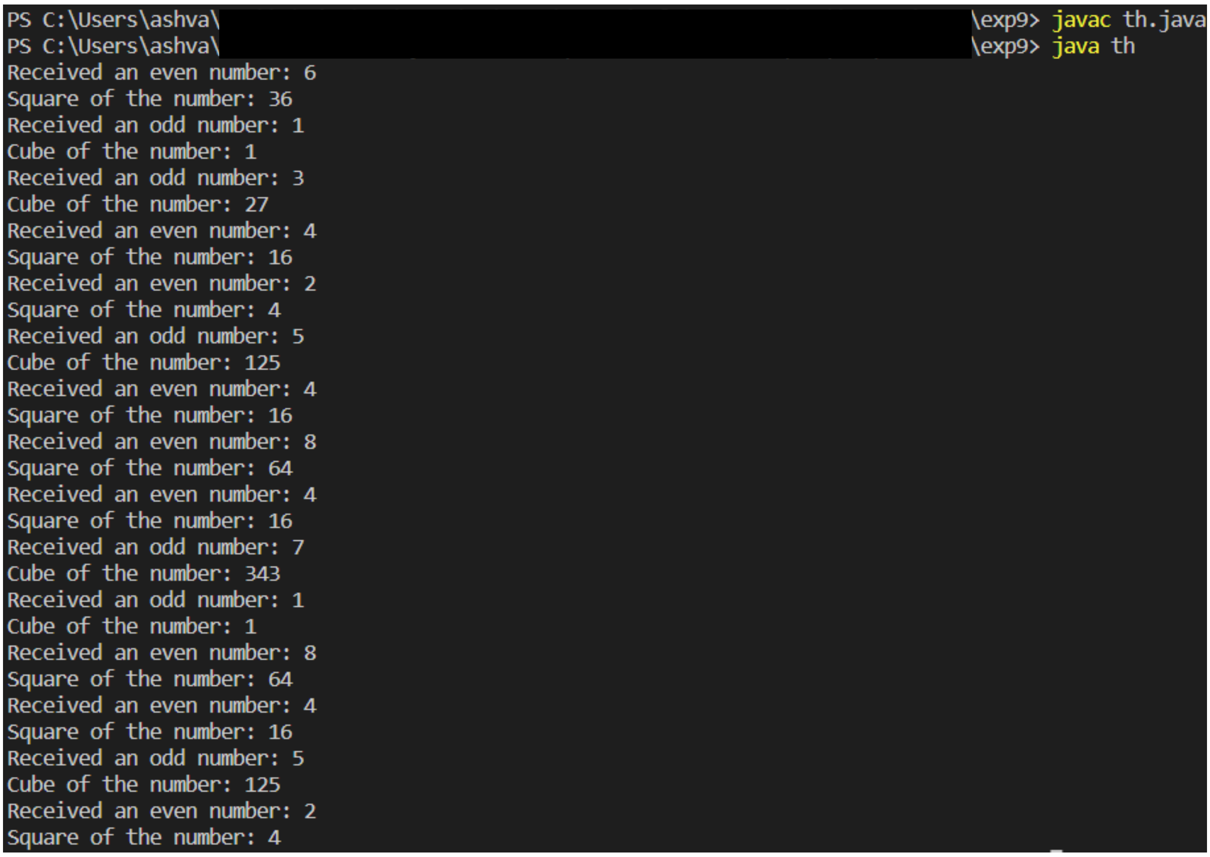
AIM:

Write a java program to implement Multithreaded programs.

PROGRAM:

```
/**
 *
 * @author 2162014
 */
import java.util.Random;

public class th {

    public static void main(String[] args) {
        Random random = new Random();
        NumberGenerator numberGenerator = new NumberGenerator(random);
        SquareCalculator squareCalculator = new SquareCalculator();
        CubeCalculator cubeCalculator = new CubeCalculator();
        Thread generatorThread = new Thread(numberGenerator);
        Thread squareThread = new Thread(squareCalculator);
        Thread cubeThread = new Thread(cubeCalculator);
        generatorThread.start();
        squareThread.start();
        cubeThread.start();
    }
}

class NumberGenerator implements Runnable {

    private final Random random;

    public NumberGenerator(Random random) {
        this.random = random;
    }

    @Override
    public void run() {
        while (true) {
            int number = random.nextInt(10);
            if (number % 2 == 0) {
                SquareCalculator.handleNumber(number);
            } else {
                CubeCalculator.handleNumber(number);
            }
            try {
                Thread.sleep(1000);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }
}

class SquareCalculator implements Runnable {

    public static synchronized void handleNumber(int number) {
        System.out.println("Received an even number: " + number);
        int square = number * number;
        System.out.println("Square of the number: " + square);
    }

    @Override
    public void run() {
        // This thread doesn't need to do anything, as the handleNumber() method
        // is static and synchronized, so it can be called from any thread.
    }
}

class CubeCalculator implements Runnable {

    public static synchronized void handleNumber(int number) {
        System.out.println("Received an odd number: " + number);
        int cube = number * number * number;
        System.out.println("Cube of the number: " + cube);
    }

    @Override
    public void run() {
        // This thread doesn't need to do anything, as the handleNumber() method
        // is static and synchronized, so it can be called from any thread.
    }
}
```

OUTPUTS:



```
PS C:\Users\ashva\[redacted]\exp9> javac th.java
PS C:\Users\ashva\\exp9> java th
Received an even number: 6
Square of the number: 36
Received an odd number: 1
Cube of the number: 1
Received an odd number: 3
Cube of the number: 27
Received an even number: 4
Square of the number: 16
Received an even number: 2
Square of the number: 4
Received an odd number: 5
Cube of the number: 125
Received an even number: 4
Square of the number: 16
Received an even number: 8
Square of the number: 64
Received an even number: 4
Square of the number: 16
Received an odd number: 7
Cube of the number: 343
Received an odd number: 1
Cube of the number: 1
Received an even number: 8
Square of the number: 64
Received an even number: 4
Square of the number: 16
Received an odd number: 5
Cube of the number: 125
Received an even number: 2
Square of the number: 4
```

RESULTS:

The java program was created successfully implements Multithreaded programs.