

**2. Develop a multithreaded Java program to create three threads. First thread generates random integer for every second and if the value is even, second thread computes the square of number and prints. If the value is odd, the third thread will print the value of cube of number.**

```
import java.util.Random;
```

```
class RandomNumberThread extends Thread {  
    public void run() {  
        Random random = new Random();  
        for (int i = 0; i < 10; i++) {  
            int randomInteger = random.nextInt(100);  
            System.out.println("Random Integer generated : " + randomInteger);  
            if((randomInteger%2) == 0) {  
                SquareThread sThread = new SquareThread(randomInteger);  
                sThread.start();  
            }  
            else {  
                CubeThread cThread = new CubeThread(randomInteger);  
                cThread.start();  
            }  
            try {  
                Thread.sleep(1000);  
            }  
            catch (InterruptedException ex) {  
                System.out.println(ex);  
            }  
        }  
    }  
}
```

```
}
```

```
class SquareThread extends Thread {
```

```
    int number;
```

```
    SquareThread(int randomNumber) {
```

```
        number = randomNumber;
```

```
    }
```

```
    public void run() {
```

```
        System.out.println("Square of " + number + " = " + (number * number));
```

```
    }
```

```
}
```

```
class CubeThread extends Thread {
```

```
    int number;
```

```
    CubeThread(int randomNumber) {
```

```
        number = randomNumber;
```

```
    }
```

```
    public void run() {
```

```
        System.out.println("Cube of " + number + " = " + number * number * number);
```

```
    }
```

```
}
```

```
public class week11p2 {
```

```
    public static void main(String args[]) {
```

```
        RandomNumberThread rnThread = new RandomNumberThread();
```

```
rnThread.start();
```

```
}
```

```
}
```