Trisha Lakhani

1BM19CS214

Thread Practice 1: Write a program to create a thread and find the sum of odd numbers from 1 to 100 in this thread. Find the sum of even numbers for the same range in the main thread.

```
import java.util.*;
import java.lang.*;


class OddThread extends Thread {
	int odd_sum=0;
	OddThread(){
		super("ODD Thread");
		System.out.println("Child Thread: "+this);
		start();
	}
	public void run(){
		try{
			for(int i=1;i<=100;i++){
				if(i%2 != 0){
					odd_sum = odd_sum + i;
					Thread.sleep(100);
				}
			}
		}catch(InterruptedException e){
			System.out.println("Child Interrupted");
		}
		System.out.println("Sum of Odd Numbers from 1 to 100: "+odd_sum);
	}
```

```
}


class ThreadPractice1 {
	public static void main(String args[]){
		int even_sum=0;
		new OddThread();
		try{
			for(int i=1;i<=100;i++){
				if(i%2 == 0){
					even_sum = even_sum + i;
					Thread.sleep(200);
				}
			}
		}catch(InterruptedException e){
			System.out.println("Main Thread Interrupted");
		}
		System.out.println("Sum of Even Numbers from 1 to 100: "+even_sum);
	}
}
```

OUTPUT

```
C:\Users\Dell\Desktop\java>javac ThreadPractice1.java

C:\Users\Dell\Desktop\java>java ThreadPractice1
Child Thread: Thread[ODD Thread,5,main]
Sum of Odd Numbers from 1 to 100: 2500
Sum of Even Numbers from 1 to 100: 2550

C:\Users\Dell\Desktop\java>
```

Thread practice 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 randomNumbern) {
            number = randomNumbern;
        }

        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 ThreadPractice2 {
        public static void main(String args[]) {
                RandomNumberThread rnThread = new RandomNumberThread();
                rnThread.start();
        }
}
```

OUTPUT

```
C:\Users\Dell\Desktop\java>javac ThreadPractice2.java

C:\Users\Dell\Desktop\java>java ThreadPractice2
Random Integer generated : 15
Cube of 15 = 3375
Random Integer generated : 31
Cube of 31 = 29791
Random Integer generated : 20
Square of 20 = 400
Random Integer generated : 91
Cube of 91 = 753571
Random Integer generated : 78
Square of 78 = 6084
Random Integer generated : 99
Cube of 99 = 970299
Random Integer generated : 22
Square of 22 = 484
Random Integer generated : 89
Cube of 89 = 704969
Random Integer generated : 92
Square of 92 = 8464
Random Integer generated : 57
Cube of 57 = 185193
```