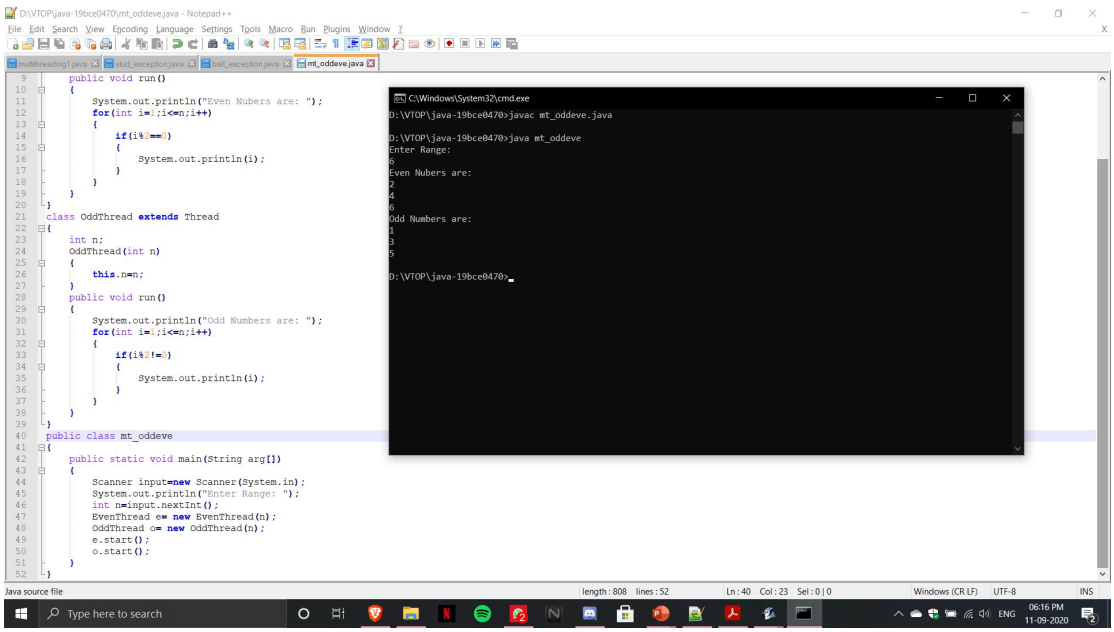
Ripunjay Narula 19BCE0470
Java Lab Digital Assignment

Multithreading:

```
1.import java.util.Scanner;
class EvenThread extends Thread
{
	int n;
	EvenThread(int n)
	{
		this.n=n;
	}
	public void run()
	{
		System.out.println("Even Nubers are: ");
		for(int i=1;i<=n;i++)
		{
			if(i%2==0)
			{
				System.out.println(i);
			}
		}
	}
}
class OddThread extends Thread
{
	int n;
	OddThread(int n)
	{
		this.n=n;
	}
	public void run()
	{
		System.out.println("Odd Numbers are: ");
		for(int i=1;i<=n;i++)
		{
			if(i%2!=0)
			{
				System.out.println(i);
			}
		}
	}
}
public class mt_oddeve
{
	public static void main(String arg[])
```

```
    {
        Scanner input=new Scanner(System.in);
        System.out.println("Enter Range: ");
        int n=input.nextInt();
        EvenThread e= new EvenThread(n);
        OddThread o= new OddThread(n);
        e.start();
        o.start();
    }
}
```



2.import java.util.*;

import java.awt.*;

class One implements Runnable

{

    One()

    {

        new Thread(this,"one").start();

```
    }

    public void run()
    {
        for(int i=0;i<6;i++)
        {
            try
            {
                Thread.sleep(1000);

            }
            catch(InterruptedException e)
            {
                System.out.println("Hello");

            }
        System.out.println("Hello");

        }
    }
}
    class Two implements Runnable
    {
        Two(){
        new Thread(this,"two").start();

    }
```

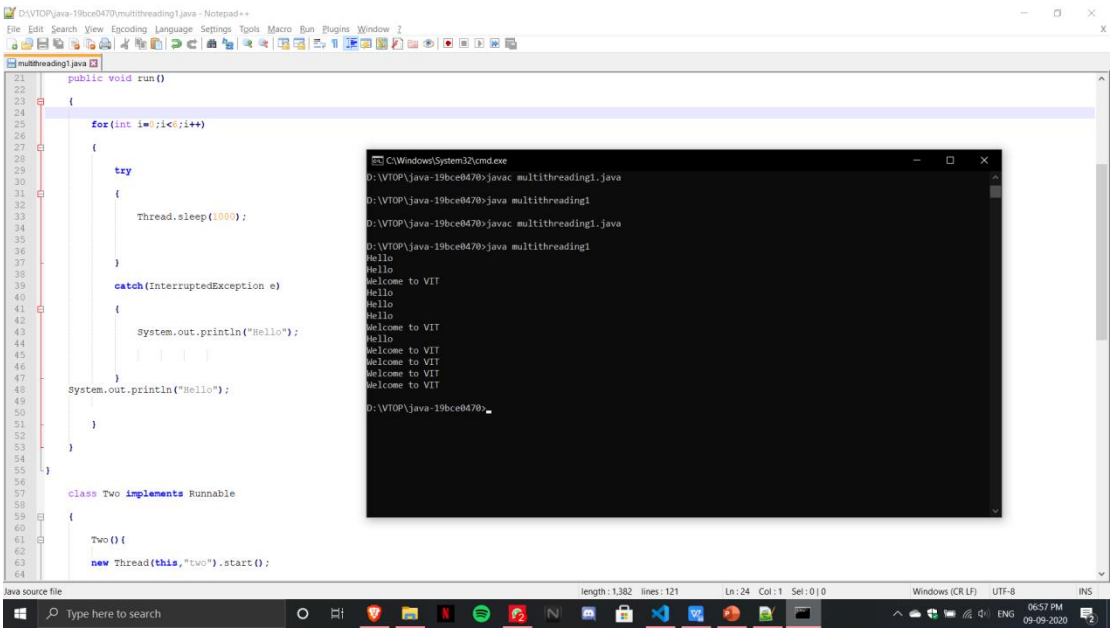
```
public void run()
{
    for(int i=0;i<6;i++)
    {
        try
        {
            Thread.sleep(3000);

        }
        catch(InterruptedException e)
        {
            System.out.println("Welcome to VIT");

        }
       System.out.println("Welcome to VIT");

    }
}
}
public class multithreading1
{
    public static void main(String args[])
    {
        One o1 = new One();
        Two t1 = new Two();
    }
```

}



3.import java.util.Random;

```
class Vote {
	int[] arr;
	int A_vote;
	int B_vote;
	int C_vote;
	public void GenerateVotes() {
		Random r = new Random();
		this.arr = new int[240];
		for (int i = 0; i < 240; i++) {
			int a = r.nextInt(3);
			this.arr[i] = a + 1;
		}
		this.A_vote = 0;
		this.B_vote = 0;
		this.C_vote = 0;
	}
	synchronized public void VoteCount(int start, int fin) {
		int A = 0, B = 0, C = 0;
		for (int i = start; i < fin; i++) {
			if (this.arr[i] == 1) {
				A++;
			} else if (this.arr[i] == 2) {
				B++;
			} else {
				C++;
			}
		}
```

```
            this.A_vote += A;
            this.B_vote += B;
            this.C_vote += C;
        }
}
class ThreadA extends Thread {
        Vote v;
        ThreadA(Vote v) {
            this.v = v;
        }
        public void run() {
            this.v.VoteCount(0, 60);
        }
}
class ThreadB extends Thread {
        Vote v;
        ThreadB(Vote v) {
            this.v = v;
        }
        public void run() {
            this.v.VoteCount(60, 120);
        }
}
class ThreadC extends Thread {
        Vote v;
        ThreadC(Vote v) {
            this.v = v;
        }
        public void run() {
            this.v.VoteCount(120, 180);
        }
}
class ThreadD extends Thread {
        Vote v;
        ThreadD(Vote v) {
            this.v = v;
        }
        public void run() {
            this.v.VoteCount(180, 240);
        }
}
public class election_mt {
        public static void main(String args[]) {
            Vote v = new Vote();
            v.GenerateVotes();
            ThreadA a = new ThreadA(v);
            ThreadB b = new ThreadB(v);
            ThreadC c = new ThreadC(v);
            ThreadD d = new ThreadD(v);
            a.start();
```

```
            b.start();
            c.start();
            d.start();
            try {
                a.join();
                b.join();
                c.join();
                d.join();
            } catch (Exception e) {
                System.out.println("Exception has " + e);
            }
            if (v.A_vote >= v.B_vote && v.A_vote >= v.C_vote) {
                System.out.println("A is the winner with " + v.A_vote + " votes");
            } else if (v.B_vote >= v.A_vote && v.B_vote >= v.C_vote) {
                System.out.println("B is the winner with " + v.B_vote + " votes");
            } else {
                System.out.println("C is the winner with " + v.C_vote + " votes");
            }
            System.out.println("A:" + v.A_vote + " B: " + v.B_vote + " C: " +
v.C_vote);
        }
    }
```