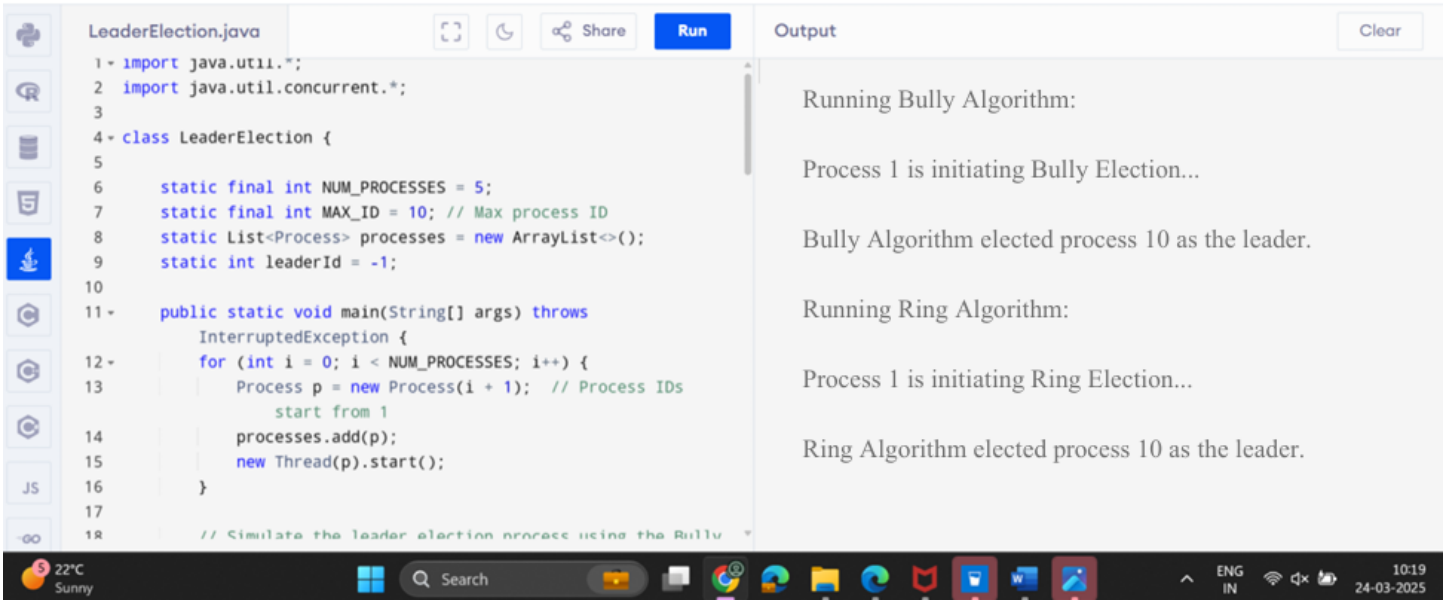
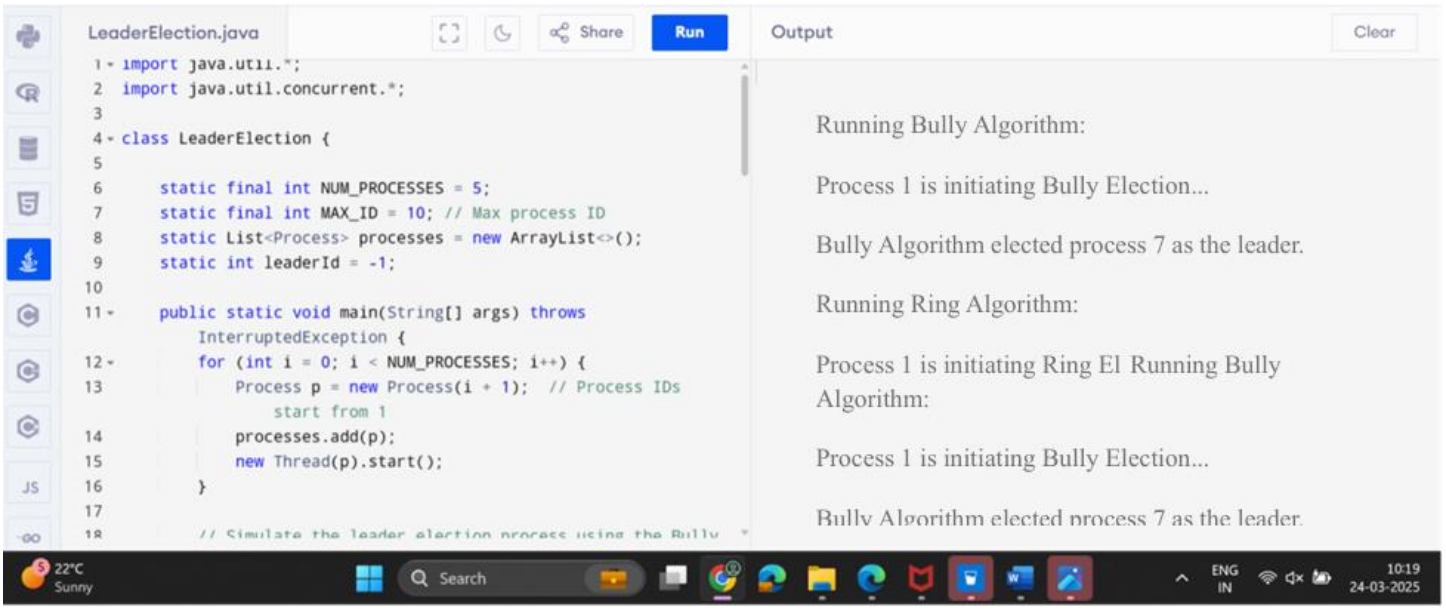
```
import java.util.*;
import java.util.concurrent.*;
class LeaderElection {
static final int NUM_PROCESSES = 5;
static final int MAX_ID = 10; // Max process ID
static List<Process> processes = new
ArrayList<>();
static int leaderId = -1;
public static void main(String[] args) throws
InterruptedException {
for (int i = 0; i < NUM_PROCESSES; i++) {
Process p = new Process(i + 1);  // Process IDs start
from 1
processes.add(p);
new Thread(p).start();
}
// Simulate the leader election process using the
Bully and Ring Algorithms
Thread.sleep(1000);
System.out.println("\nRunning Bully Algorithm:");
runBullyAlgorithm();
Thread.sleep(2000);
System.out.println("\nRunning Ring Algorithm:");
runRingAlgorithm();
}
static class Process implements Runnable {
int id;
1 DS-6 code
boolean isAlive = true;
Random rand = new Random();
Process(int id) {
this.id = id;
}
@Override
public void run() {
try {
while (isAlive) {
Thread.sleep(rand.nextInt(5000));
if (leaderId == -1) {
leaderId = this.id; // Randomly assign a leader for
the first time
}
}
} catch (InterruptedException e) {
e.printStackTrace();
}
}
// Initiates the Bully Algorithm for leader election
static void runBullyAlgorithm() throws
InterruptedException {
System.out.println("Process " + processes.get(0).id
+ " is initiating Bully Election...");
int maxId = -1;
for (Process process : processes) {
if (process.id > maxId) {
maxId = process.id;
}
}
leaderId = maxId;
2 DS-6 code
System.out.println("Bully Algorithm elected
process " + leaderId + " as the leader.");
```

```
}
// Runs the Ring Algorithm for leader election
static void runRingAlgorithm() throws InterruptedException {
System.out.println("Process " + processes.get(0).id + " is initiating Ring Election...");
int highestId = -1;
for (Process process : processes) {
if (process.id > highestId) {
highestId = process.id;
}
}
leaderId = highestId;
System.out.println("Ring Algorithm elected process " + leaderId + " as the leader.");
}
}
}
```

3 DS-6 code

Output :





LeaderElection.java

Share Run

```java
import java.util.*;
import java.util.concurrent.*;

class LeaderElection {

    static final int NUM_PROCESSES = 5;
    static final int MAX_ID = 10; // Max process ID
    static List<Process> processes = new ArrayList<>();
    static int leaderId = -1;

    public static void main(String[] args) throws
        InterruptedException {
        for (int i = 0; i < NUM_PROCESSES; i++) {
            Process p = new Process(i + 1);  // Process IDs
                start from 1
            processes.add(p);
            new Thread(p).start();
        }

```

Output Clear

```
Running Bully Algorithm:

Process 1 is initiating Bully Election...

Bully Algorithm elected process 9 as the leader.

Running Ring Algorithm:

Process 1 is initiating Ring El Running Bully Algorithm:

Process 1 is initiating Bully Election...

Bully Algorithm elected process 9 as the leader.
```