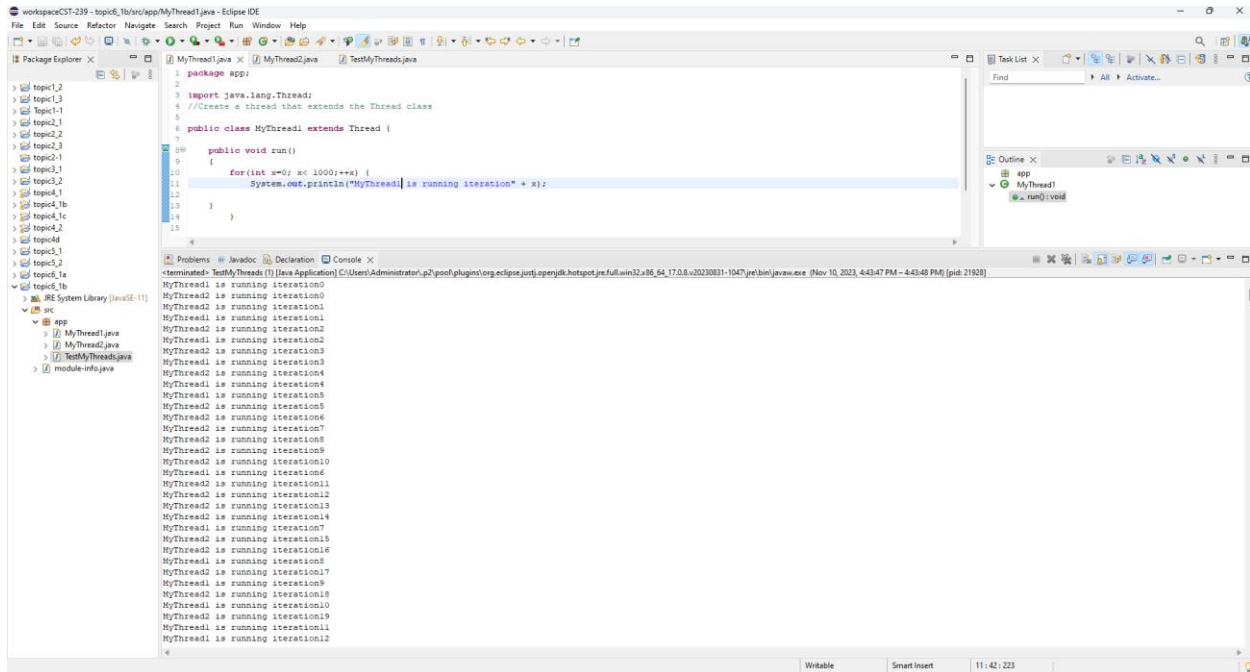


# Activity 6-1b

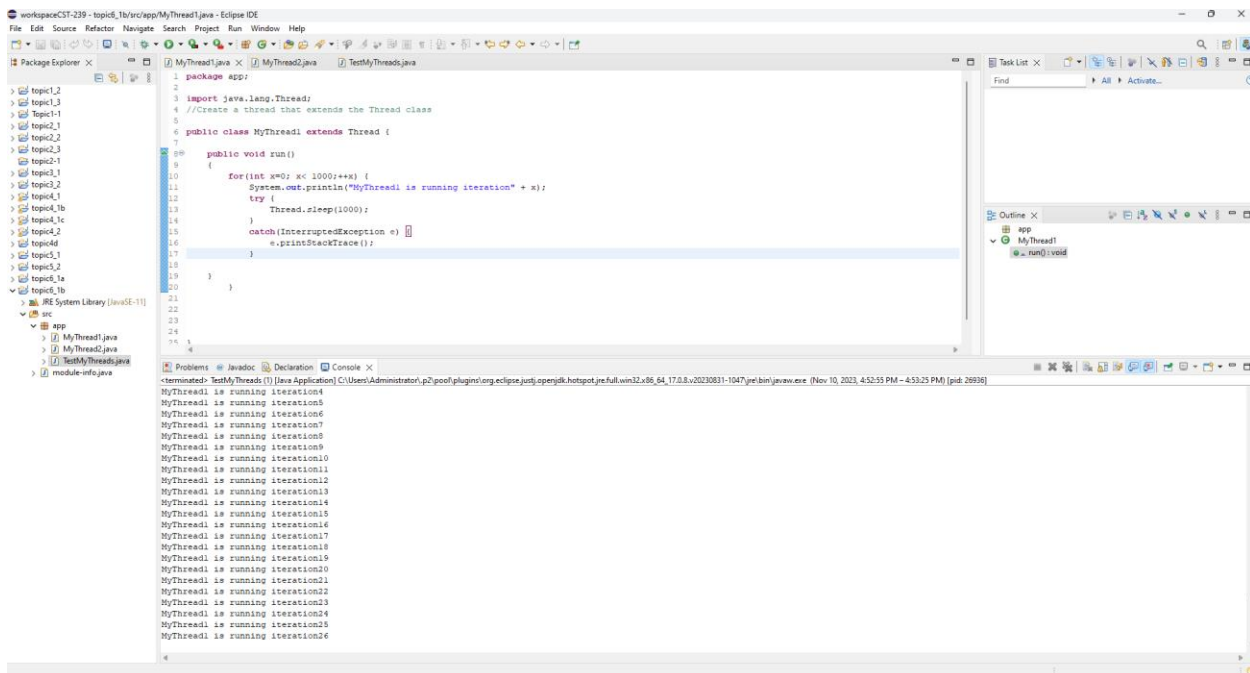
## ScreenShot of output:



```
1 package app;
2
3 import java.lang.Thread;
4 //Create a thread that extends the Thread class
5
6 public class MyThread1 extends Thread {
7
8     public void run()
9     {
10         for(int x=0; x< 1000; x++) {
11             System.out.println("MyThread1 is running iteration" + x);
12         }
13     }
14 }
15
```

Console Output:

```
<terminated> TestMyThreads (1) [Java Application] C:\Users\Administrator\p2\pool\plugins\org.eclipse.jdt.launcher\org.eclipse.jdt.launcher.exe (Nov 10, 2023, 4:43:47 PM - 4:43:48 PM) [pid: 21928]
MyThread1 is running iteration0
MyThread2 is running iteration0
MyThread2 is running iteration1
MyThread1 is running iteration1
MyThread2 is running iteration2
MyThread1 is running iteration2
MyThread2 is running iteration3
MyThread1 is running iteration3
MyThread2 is running iteration4
MyThread1 is running iteration4
MyThread2 is running iteration5
MyThread1 is running iteration5
MyThread2 is running iteration6
MyThread1 is running iteration6
MyThread2 is running iteration7
MyThread1 is running iteration7
MyThread2 is running iteration8
MyThread1 is running iteration8
MyThread2 is running iteration9
MyThread1 is running iteration9
MyThread2 is running iteration10
MyThread1 is running iteration10
MyThread2 is running iteration11
MyThread1 is running iteration11
MyThread2 is running iteration12
MyThread1 is running iteration12
```

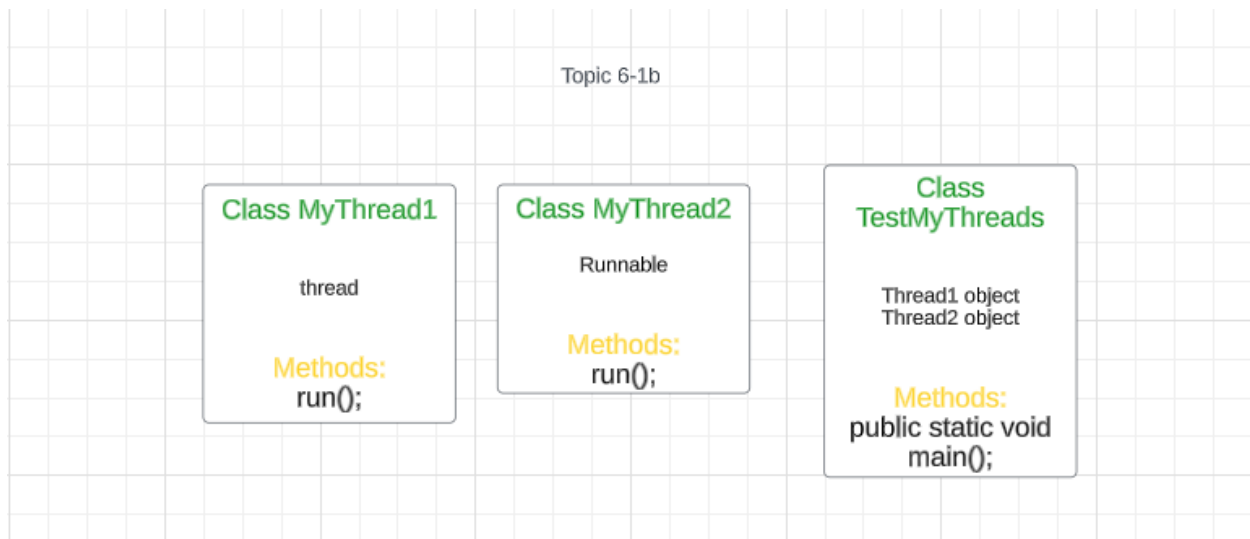


```
1 package app;
2
3 import java.lang.Thread;
4 //Create a thread that extends the Thread class
5
6 public class MyThread1 extends Thread {
7
8     public void run()
9     {
10         for(int x=0; x< 1000; x++) {
11             System.out.println("MyThread1 is running iteration" + x);
12             try {
13                 Thread.sleep(1000);
14             } catch (InterruptedException e) {}
15             e.printStackTrace();
16         }
17     }
18 }
19
20
21
22
23
24
25
26
```

Console Output:

```
<terminated> TestMyThreads (1) [Java Application] C:\Users\Administrator\p2\pool\plugins\org.eclipse.jdt.launcher\org.eclipse.jdt.launcher.exe (Nov 10, 2023, 4:52:55 PM - 4:53:25 PM) [pid: 26936]
MyThread1 is running iteration4
MyThread1 is running iteration5
MyThread1 is running iteration6
MyThread1 is running iteration7
MyThread1 is running iteration8
MyThread1 is running iteration9
MyThread1 is running iteration10
MyThread1 is running iteration11
MyThread1 is running iteration12
MyThread1 is running iteration13
MyThread1 is running iteration14
MyThread1 is running iteration15
MyThread1 is running iteration16
MyThread1 is running iteration17
MyThread1 is running iteration18
MyThread1 is running iteration19
MyThread1 is running iteration20
MyThread1 is running iteration21
MyThread1 is running iteration22
MyThread1 is running iteration23
MyThread1 is running iteration24
MyThread1 is running iteration25
MyThread1 is running iteration26
```

## UML:



## Write up:

### 6-1b, part 1 write up

we had the same code as activity 6-1a, which consisted of 2 threads and another class that implements and extends off that main class. The `myThread1` has a loop that iterated 1000 times, while `mythread2` did that in 100 times. When I ran the code, I was expecting for one of the threads to finish the loop before the other one starts. My output shows different, where it was running them randomly. Thread 1 running and then thread2 running so they were both running at the same time instead one by one.

### 6-1b, part 2 write up

We have the same code still; except we have added a thread sleeper which makes the thread suspend its execution time. So, the output was different this time because now we have one thread running one by one instead of both threads running sequentially. So, my output was showing a thread by a thread, without having the second thread show its output. So, the sleeper method made one of the threads suspend its output.