

JAVA PROGRAMMING

LA - 5

Q1 Write a synchronization program for the ticket booking process in the theatre. Consider three counters for booking and have `num_tickets` as a common variable holding available tickets. When counter1 is using it others should not access variable `num_tickets`.

CODE:

```
import java.util.Scanner;

class TicketBookingSystem {

    private static int numTickets;

    public static void main(String[] args) {

        System.out.println("Shifa Khan, 21BBS0255");

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the number of available tickets: ");
        numTickets = scanner.nextInt();

        Counter counter1 = new Counter("Counter 1");
        Counter counter2 = new Counter("Counter 2");
        Counter counter3 = new Counter("Counter 3");

        Thread thread1 = new Thread(counter1);
        Thread thread2 = new Thread(counter2);
        Thread thread3 = new Thread(counter3);

        thread1.start();
        thread2.start();
        thread3.start();
    }

    static class Counter implements Runnable {

        private String counterName;
```

```

public Counter(String counterName) {

    this.counterName = counterName;

}

@Override

public void run() {

    while (true) {

        synchronized (TicketBookingSystem.class) {

            if (numTickets > 0) {

                System.out.println(counterName + " is booking a ticket. Available tickets: " + numTickets);

                numTickets--;

                try {

                    Thread.sleep(1000);

                } catch (InterruptedException e) {

                    e.printStackTrace();

                }

            } else {

                System.out.println(counterName + " has no more tickets to book. Exiting.");

                break;

            }

        }

    }

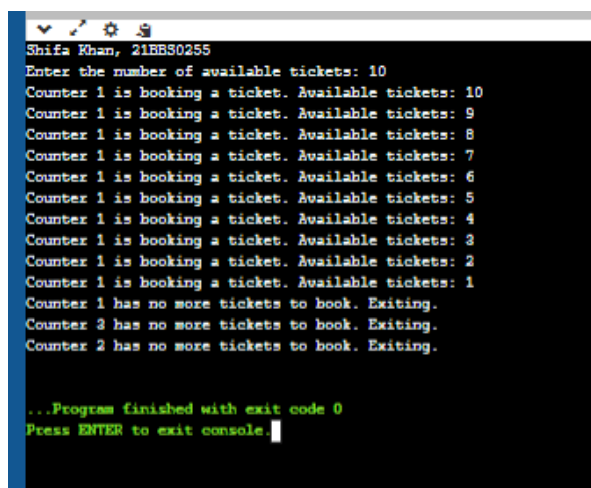
}

}

}

```

OUTPUT:



```

Shifa_Khan, 21BES0255
Enter the number of available tickets: 10
Counter 1 is booking a ticket. Available tickets: 10
Counter 1 is booking a ticket. Available tickets: 9
Counter 1 is booking a ticket. Available tickets: 8
Counter 1 is booking a ticket. Available tickets: 7
Counter 1 is booking a ticket. Available tickets: 6
Counter 1 is booking a ticket. Available tickets: 5
Counter 1 is booking a ticket. Available tickets: 4
Counter 1 is booking a ticket. Available tickets: 3
Counter 1 is booking a ticket. Available tickets: 2
Counter 1 is booking a ticket. Available tickets: 1
Counter 1 has no more tickets to book. Exiting.
Counter 3 has no more tickets to book. Exiting.
Counter 2 has no more tickets to book. Exiting.

...Program finished with exit code 0
Press ENTER to exit console.

```

SCREENSHOT:

Online C Compiler - online edit

https://www.onlinegdb.com/online_c_compiler#

VOLSWIFI Authentic... Plagiarism Checker... Harish Note: K & R... Problems - LeetCode GitHub Dashboard | Hacker... Forests and their M... VIT Placement VIT | FACE Prep Dashboard - Vercel All Bookmarks

Run Debug Stop Save Ready

Language Java

```
1  import java.util.Scanner;
2
3  class TicketBookingSystem {
4      private static int numTickets;
5
6      public static void main(String[] args) {
7
8          System.out.println("Shifa Khan, 218858255");
9
10         Scanner scanner = new Scanner(System.in);
11
12         System.out.print("Enter the number of available tickets: ");
13         numTickets = scanner.nextInt();
14
15         Counter counter1 = new Counter("Counter 1");
16         Counter counter2 = new Counter("Counter 2");
17         Counter counter3 = new Counter("Counter 3");
18
19         Thread thread1 = new Thread(counter1);
20         Thread thread2 = new Thread(counter2);
21         Thread thread3 = new Thread(counter3);
22
23         thread1.start();
24         thread2.start();
25         thread3.start();
26
27     }
28
29     static class Counter implements Runnable {
30         private String counterName;
31
32         public Counter(String counterName) {
33             this.counterName = counterName;
34         }
35
36         @Override
37         public void run() {
38             while (true) {
39                 synchronized (TicketBookingSystem.class) {
40                     if (numTickets > 0) {
41                         System.out.println(counterName + " is booking a ticket. Available tickets: " + numTickets);
42                         numTickets--;
43
44                         try {
45                             Thread.sleep(1000);
46                         } catch (InterruptedException e) {
47                             e.printStackTrace();
48                         }
49                     } else {
50                         System.out.println(counterName + " has no more tickets to book. Exiting.");
51                         break;
52                     }
53                 }
54             }
55         }
56     }
57 }
58
```

Input

Enter the number of available tickets: 10
Counter 1 is booking a ticket. Available tickets: 10
Counter 1 is booking a ticket. Available tickets: 9
Counter 1 is booking a ticket. Available tickets: 8
Counter 1 is booking a ticket. Available tickets: 7
Counter 1 is booking a ticket. Available tickets: 6
Counter 1 is booking a ticket. Available tickets: 5
Counter 1 is booking a ticket. Available tickets: 4
Counter 1 is booking a ticket. Available tickets: 3
Counter 1 is booking a ticket. Available tickets: 2
Counter 1 is booking a ticket. Available tickets: 1

Taskbar

Windows Start Search Type here to search Task View L2 Edge File Explorer Google Chrome VS Code IntelliJ IDEA Docker Desktop Word 9:29 27°C Clear 1903 12-03-2024