

## Code-

```
import java.util.*;

class TicketBookingSystem {
    private final int totalSeats;
    private final boolean[] seats;

    public TicketBookingSystem(int totalSeats) {
        this.totalSeats = totalSeats;
        this.seats = new boolean[totalSeats];
    }

    public synchronized boolean bookSeat(int seatNumber, String user) {
        if (seatNumber < 0 || seatNumber >= totalSeats || seats[seatNumber]) {
            System.out.println(user + " - Seat " + seatNumber + " is already booked or invalid.");
            return false;
        }
        seats[seatNumber] = true;
        System.out.println(user + " booked seat " + seatNumber);
        return true;
    }

    public synchronized void printSeats() {
        System.out.println("Current Seat Status:");
        for (int i = 0; i < totalSeats; i++) {
            System.out.println("Seat " + i + ": " + (seats[i] ? "Booked" : "Available"));
        }
    }
}

class BookingThread extends Thread {
    private final TicketBookingSystem system;
    private final String user;
    private final int seatNumber;

    public BookingThread(TicketBookingSystem system, String user, int seatNumber, int
        priority) {
        this.system = system;
        this.user = user;
        this.seatNumber = seatNumber;
        setPriority(priority);
    }

    @Override
    public void run() {
        system.bookSeat(seatNumber, user);
    }
}
```

```

}

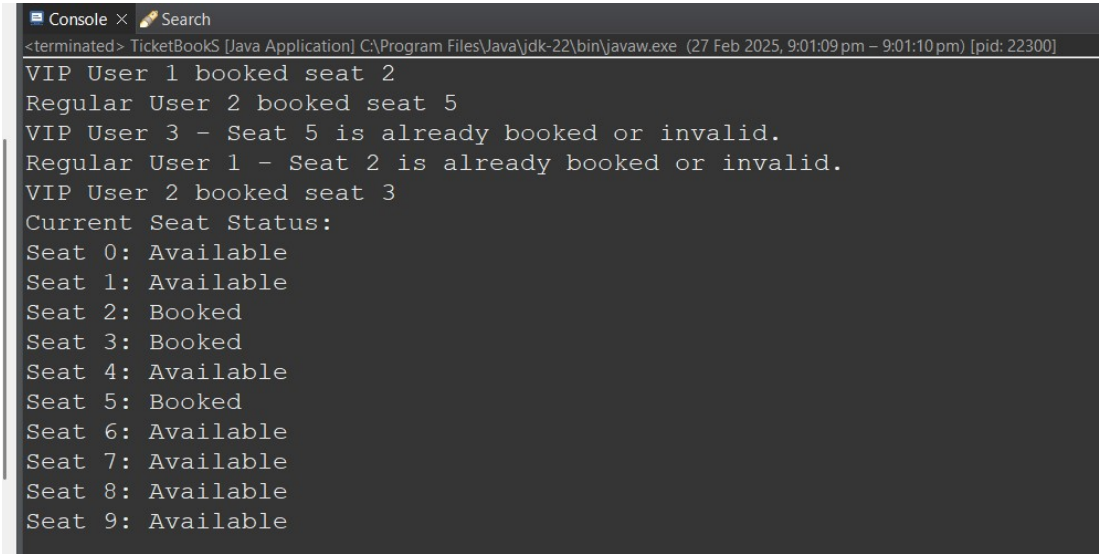
public class TicketBookS {
    public static void main(String[] args) {
        TicketBookingSystem system = new TicketBookingSystem(10);
        List<Thread> threads = new ArrayList<>();

        threads.add(new BookingThread(system, "VIP User 1", 2, Thread.MAX_PRIORITY));
        threads.add(new BookingThread(system, "VIP User 2", 3, Thread.MAX_PRIORITY));
        threads.add(new BookingThread(system, "Regular User 1", 2,
            Thread.MIN_PRIORITY));
        threads.add(new BookingThread(system, "Regular User 2", 5,
            Thread.NORM_PRIORITY));
        threads.add(new BookingThread(system, "VIP User 3", 5, Thread.MAX_PRIORITY));

        for (Thread t : threads) {
            t.start();
        }
        for (Thread t : threads) {
            try {
                t.join();
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
        system.printSeats();
    }
}

```

## Input/ Output



```

<terminated> TicketBookS [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe (27 Feb 2025, 9:01:09 pm - 9:01:10 pm) [pid: 22300]
VIP User 1 booked seat 2
Regular User 2 booked seat 5
VIP User 3 - Seat 5 is already booked or invalid.
Regular User 1 - Seat 2 is already booked or invalid.
VIP User 2 booked seat 3
Current Seat Status:
Seat 0: Available
Seat 1: Available
Seat 2: Booked
Seat 3: Booked
Seat 4: Available
Seat 5: Booked
Seat 6: Available
Seat 7: Available
Seat 8: Available
Seat 9: Available

```

