

Experiment - 4.3

Student Name - Sneha Gupta

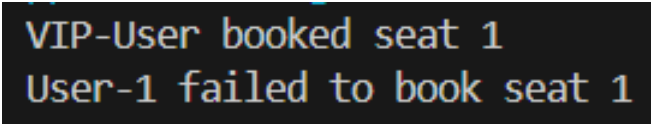
UID - 22BCS11233

Aim: Develop a ticket booking system with synchronized threads to ensure no double booking of seats. Use thread priorities to simulate VIP bookings being processed first.

```
Code: import java.util.ArrayList;
import java.util.List;
class Seat {
    int seatNumber;
    boolean isBooked;
    Seat(int seatNumber) {
        this.seatNumber = seatNumber;
        this.isBooked = false;
    }
}
class TicketBooking implements Runnable {
    List<Seat> seats;
    int seatToBook;
    TicketBooking(List<Seat> seats, int seatToBook) {
        this.seats = seats;
        this.seatToBook = seatToBook;
    }
    public void run() {
        synchronized (seats) {
            for (Seat seat : seats) {
                if (seat.seatNumber == seatToBook && !seat.isBooked) {
                    seat.isBooked = true;
                    System.out.println(Thread.currentThread().getName() + " booked seat " +
seatToBook);
                    return;
                }
            }
            System.out.println(Thread.currentThread().getName() + " failed to book seat " +
seatToBook);
        }
    }
}
public class TicketBookingSystem {
    public static void main(String[] args) {
        List<Seat> seats = new ArrayList<>();
```

```
for (int i = 1; i <= 5; i++) {  
    seats.add(new Seat(i));  
}  
Thread t1 = new Thread(new TicketBooking(seats, 1), "User-1");  
Thread t2 = new Thread(new TicketBooking(seats, 1), "VIP-User");  
t1.setPriority(Thread.MIN_PRIORITY);  
t2.setPriority(Thread.MAX_PRIORITY);  
t2.start();  
t1.start();  
}  
}
```

OUTPUT



```
VIP-User booked seat 1  
User-1 failed to book seat 1
```