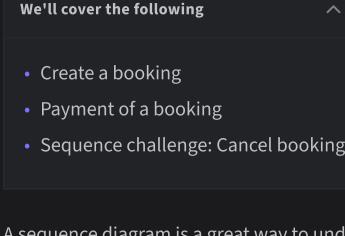
Sequence Diagram for the Movie Ticket Booking System Visualize the sequence diagrams for the creation and payment of a booking, and practice the concepts with

a challenge.



can create for our movie ticket booking system. In this lesson, we will create sequence diagrams for the following three interactions: • **Create a booking:** The customer creates a booking for a show. • Payment for the booking: The customer pays for the booking.

• **Sequence challenge:** The customer cancels their booking.

- Create a booking
- The sequence diagram for creating a booking should have the following actors and

• Actor: Customer

• **Objects:** Catalog, Movie, and ShowTime System

- Here're the steps of the interaction to create a booking interaction:
 - 1. The customer searches for a movie from the catalog.

2. The catalog returns the required movie.

objects that will interact with each other:

3. They request show times for the movie they wanted to book. 4. The available show times for the movie are returned.

I. The customer receives a list of available seats.

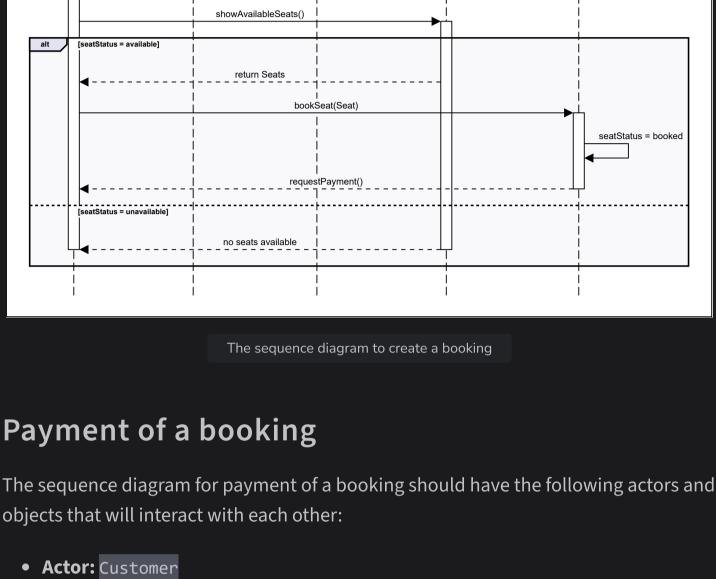
- 5. The customer requests available seats for the show. 6. If seats are available:
- II. The customer books their desired seats using the system. III. The seat status is set to book.
- 7. Else if the seats are not available: I. The customer is informed that no seats are available.

ticket booking system is given below.

sd create booking

- Based on the order above, the sequence diagram for creating a booking in the movie

IV. The system requests payment against the booked seats from the customer.



Here are the steps in the payment of a booking interaction:

II. The system creates an object for the booking.

III. The booking status is updated and the system is informed.

3. If the payment is complete: I. The payment informs the customer about the completed payment.

information.

sd booking payment

initiatePayment(bookingFee)

Payment

4. Else if the payment is declined:

Object: Payment, Seat

System

- II. The system requests to revert the seat status. III. The seat status is updated to available and returned to the system.

System

createBooking()

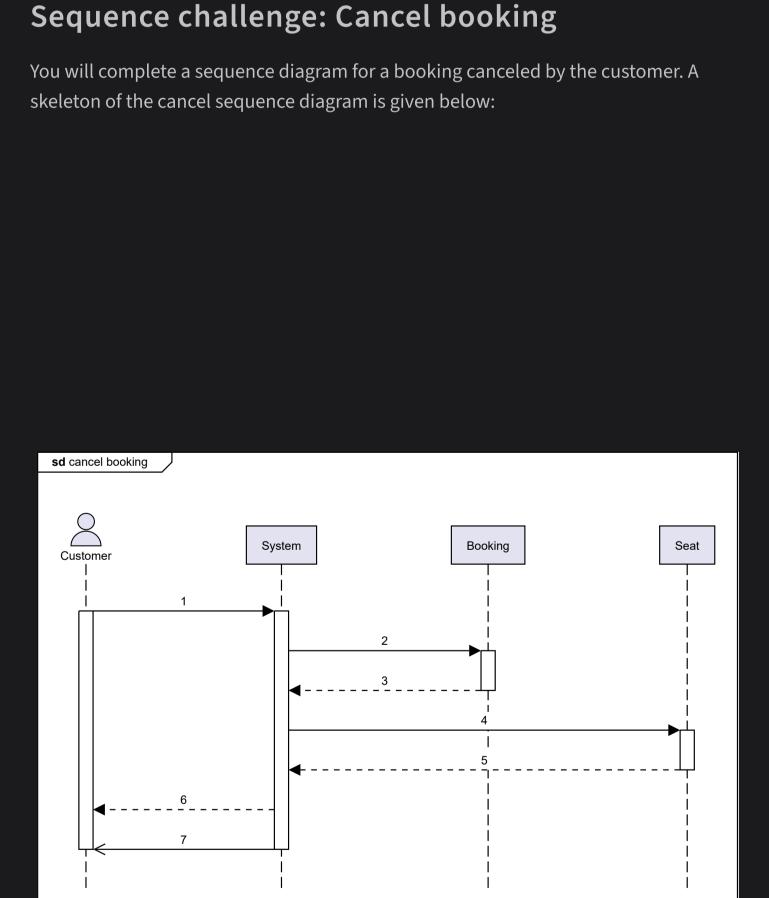
booking created

Booking

bookingStatus = confirmed

Based on the order above, the sequence diagram for payment of a booking in the movie ticket booking system is given below.

IV. The system informs the customer that the booking is declined.



updateSeatStatus(available) seatStatus updated

Alternatively, you can also click the "Show complete diagram" button below to see the complete sequence diagram for the cancel booking interaction. Hide complete diagram sd cancel booking System **Booking** Seat cancelBooking() updateBookingStatus(canceled)

Reset **Show Solution**

sendNotification(cancelationDet ails)

 Sequence challenge: Cancel booking A sequence diagram is a great way to understand the interactions between different entities and objects in the system. There can be different sequence diagrams that we

Catalog ShowTime Movie System searchMovie() getShowTime() return ShowTime

1. The customer initiates a payment against the booking fee. 2. The payment processes the payment and informs the system.

IV. The system sends a notification to the customer containing the booking

I. The payment informs the customer about the declined payment.

payment performed

payment declined revert seatStatus booking declined The sequence diagram for the payment of a booking interaction

Notice that the arrows in the diagram above are numbered from 1 to 7. The message boxes shown below are the messages to be exchanged between the actor(s) and object(s). Can you rearrange the messages below in the correct sequence of order they should appear in the skeleton of the sequence diagram given above?

Note: If you get stuck, just click the "Show Solution" button to check the

cancelBooking()

updateBookingStatus(canceled)

bookingStatus updated

correct answer.

Rearrange the sequence.

佪

The sequence diagram for canceling the booking

- ______ refund payment
- bookingStatus updated