CS250 UML Diagram Relations

The UML Diagram describes classes and how they relate to one another for a ticketing management system which could be used by a movie theater. It includes the following classes:

**Theater:** has private members which represent all attributes of a movie theater including **name**, **location**, **phone**, **max capacity**, and **closing and opening times**. It has public methods which can retrieve lists of movies and showtimes, as well as one to reserve seats and another to cancel reservations made.

**Movie:** represents a movie and has members unique to each movie, such as, **movieTitle**, **movieDuration**, description, and criticRating. It has 2 public methods (**getMovieInfo()** & **getAvailaibility(movie)**) which are able to fetch information about a movie and check for ticket availability on a movie respectively.

**Ticket:** holds information pertaining to a ticket with private members such as: **ticketAvailability**, **ticketID**, **ticketCost**, **movie**, and **seat** (of type seating). It has public methods to check if a ticket is available, and confirm whether it is a valid existing ticket.

**Order:** responsible for storing and managing attributes of an order such as the ticket & information with the private members being: **orderID**, **showing**, **showingDate**, **showingTime**, **customer**, **ticketID**, and **totalCost.** Has 3 public methods capable of calculating your order total, with the optional addition of a discount code, and lastly a boolean returning whether the order was successfully placed and received.

**Customer:** represents a customer with private members **name**, **phone number**, and **email.**

**Showing:** represents a specific date and time for a showing of a movie, with private member variables **date**, **time**, and **movie** (of type Movie)

**Seating:** manages the ordering and purchases of the seats for a given showtime, with private members being **rowLetter**, **seatNumber**, **seatType.**

**Payment:** responsible for implementing payments for a movie ticket. Has **paymentId**, **totalCost** (before tax), and **paymentStatus** members to indicate its current state

**Class Relations**

**Theater→ Showing: the** theater can have multiple showings of different movies, however since this is only for one theater there cannot be showings for other theaters.

**Movie→Tickets**: Each ticket is for a specific movie and there can be multiple, however there will never be movies playing at the theater with no tickets available (under normal circumstances).

**Showing→ Seating**: Each showing will have a specific number of seats available for purchase.

**Tickets→ Showing & Seating:** Each ticket corresponds to a specific showtime and specific seats which are reserved.

**Order→Tickets**: An order will always contain tickets, and can contain multiple.

**Order→Customer**: Each order has one customer

**Customer→Order→Payment**: Customer can place an order for tickets which will then require payment which is also associated with each order (every order requires payment)

**Payment→Order**: Each payment is for a specific order; don't want to accidentally purchase someone else's movie tickets with their seats and showtime.