

Sitzplatzreservierung für einen Theatersaal

Erstelle ein C#-Programm, das eine Sitzplatzreservierung für einen Theatersaal ermöglicht. Das Programm soll es dem Benutzer erlauben, Sitzplätze zu reservieren, Reservierungen zu stornieren und Tickets zu bestellen.

Klassen: Seat, Theater, Ticket

Klasse: Seat

Eigenschaften:

Row (int): Die Reihe des Sitzplatzes.

Number (int): Die Nummer des Sitzplatzes in der jeweiligen Reihe.

IsReserved (bool): Gibt an, ob der Sitzplatz reserviert ist.

IsSold (bool): Gibt an, ob der Sitzplatz verkauft wurde.

Konstruktor:

Erstellt einen Sitzplatz mit der angegebenen Reihe und Nummer. Der Sitzplatz ist standardmäßig nicht reserviert und nicht verkauft.

Methoden:

Reserve(): Reserviert den Sitzplatz, falls er noch nicht reserviert ist. Gibt keinen Rückgabewert zurück.

CancelReservation(): Hebt die Reservierung auf, falls der Sitzplatz reserviert ist. Gibt keinen Rückgabewert zurück.

Sell(): Markiert den Sitzplatz als verkauft. Gibt keinen Rückgabewert zurück. Nur reservierte Plätze können verkauft werden.

ToString(): Gibt eine Zeichenkette zurück, die den Status des Sitzplatzes anzeigt: 'O' für frei, '*' für reserviert und 'X' für verkauft.

Klasse: Theater

Eigenschaften:

Name (string): Der Name des Theatersaals.

RowCount (int): Die Anzahl der Reihen im Saal.

SeatsPerRow (int): Die Anzahl der Sitzplätze pro Reihe.

Seats (Seat[,]): Speichert den Sitzplatz an der richtigen Position in einem zweidimensionalen Array.

Konstruktor:

Erstellt einen Theatersaal mit dem angegebenen Namen, einer Anzahl von Reihen und einer Anzahl von Sitzplätzen pro Reihe. Initialisiert die Sitzplätze im Array entsprechend.

Methoden:

ReserveSeat(int row, int number): Reserviert den Sitzplatz mit der angegebenen Reihe und Nummer, falls verfügbar. Gibt keinen Rückgabewert zurück.

CancelSeat(int row, int number): Storniert die Reservierung des angegebenen Sitzplatzes, falls er reserviert ist. Gibt keinen Rückgabewert zurück.

ShowAvailableSeats(): Zeigt alle verfügbaren Sitzplätze im Saal an. Gibt keinen Rückgabewert zurück.

ShowSeatingChart(): Gibt eine übersichtliche Darstellung des Sitzplans im Theatersaal aus, wobei die Sitzplätze in Spalten (Sitzplatznummern) und Reihen (Reihennummern) angeordnet sind. Verwendete Symbole:

- 'O': Frei (nicht reserviert)
- '*': Reserviert
- 'X': Verkauft und Beahlt. Gibt keinen Rückgabewert zurück.

SellSeat(int row, int number): Versucht, den angegebenen Sitzplatz zu verkaufen. Gibt ein Ticket-Objekt zurück, wenn der Verkauf erfolgreich war, andernfalls wird ein Fehler ausgelöst.

Klasse: Ticket

Eigenschaften:

Seat (Seat): Der reservierte Sitzplatz, für den das Ticket ausgestellt wird.

Price (decimal): Der Preis für das Ticket.

Konstruktor:

Erstellt ein Ticket für den angegebenen Sitzplatz mit dem angegebenen Preis. Der Sitzplatz muss reserviert sein, bevor ein Ticket bestellt werden kann.

Methoden:

Order(): Gibt eine Bestätigung aus, dass das Ticket bestellt wurde. Wenn der Sitzplatz nicht reserviert ist, wird eine Fehlermeldung ausgegeben. Gibt keinen Rückgabewert zurück.

Anforderungen

- Benutzer sollen Sitzplätze reservieren und Reservierungen stornieren können.
- Es soll eine Methode geben, um die verfügbaren Sitzplätze im Theatersaal anzuzeigen.
- Es soll möglich sein, Tickets für bereits reservierte Sitzplätze zu bestellen.
- Fehlerbehandlung: Wenn ein Sitzplatz bereits reserviert ist oder nicht existiert, soll eine aussagekräftige Fehlermeldung ausgegeben werden.

Main-Methode:

Eine mögliche Main-Methode könnte wie folgt aussehen, wobei die zufällige Reservierungsmethode als statische Methode innerhalb der Klasse Program selbst geschrieben werden müsste:

```
public class Program {
    public static void Main(string[] args) {
        // Erstelle einen Theatersaal mit einem Einheitspreis von 30 Euro
        Theater theater = new Theater("Stadttheater", 5, 10, 30.00m);
        // Zeige den Sitzplan an
        theater.ShowSeatingChart();
        // Reserviere einige Sitzplätze
        theater.ReserveSeat(1, 1);
        theater.ReserveSeat(2, 5);
        // Zeige den aktualisierten Sitzplan an
        Console.WriteLine("\nNach der Reservierung:");
        theater.ShowSeatingChart();
        // Storniere eine Reservierung
        theater.CancelSeat(1, 1);
        // Zeige den Sitzplan nach der Stornierung an
        Console.WriteLine("\nNach der Stornierung:");
        theater.ShowSeatingChart();
        // Versuche, einen reservierten Sitzplatz zu verkaufen
        Ticket ticket = theater.SellSeat(2, 5);
        Console.WriteLine($"Ticket verkauft für Reihe {ticket.Seat.Row + 1},
                           Sitz {ticket.Seat.Number + 1}. Preis: {ticket.Price} Euro");
        // Freie Sitzplätze anzeigen
        theater.ShowAvailableSeats();
        // Reserviere zufällige Sitzplätze
        ReserveRandomSeats(theater, 50); // Versuche, 50 zufällige Reservierungen
        // Zeige den endgültigen Sitzplan an
        Console.WriteLine("\nEndgültiger Sitzplan:");
        theater.ShowSeatingChart();
    }

    // Statische Methode zum Reservieren zufälliger Sitzplätze
    public static void ReserveRandomSeats(Theater theater, int numberOfRes) {...}
```

Klassendiagramm:

Program

Klasse

Methoden

- Main(string[] args) : void
- ReserveRandomSeats(Theater theater, int numberOfReservations) : v...

Seat

Klasse

Eigenschaften

- IsReserved { get; set; } : bool
- IsSold { get; set; } : bool
- Number { get; } : int
- Row { get; } : int

Methoden

- CancelReservation() : void
- Reserve() : void
- Seat(int row, int number)
- Sell() : void
- ToString() : string

Ticket

Klasse

Eigenschaften

- Price { get; } : decimal
- Seat { get; } : Seat

Methoden

- Ticket(Seat seat, decimal price)

Theater

Klasse

Felder

- TicketPrice : decimal

Eigenschaften

- Name { get; } : string
- RowCount { get; } : int
- Seats { get; } : Seat[,]
- SeatsPerRow { get; } : int

Methoden

- CancelSeat(int row, int number) : void
- ReserveSeat(int row, int number) : void
- SellSeat(int row, int number) : Ticket
- ShowAvailableSeats() : void
- ShowSeatingChart() : void
- Theater(string name, int rowCount, int seatsPerRow, decimal ticketPrice)