

Train Seat Reservation System

Problem Description:

1. There are 80 seats in a coach of a train with only 7 seats in a row and last row of only 3 seats. For simplicity, there is only one coach in this train.
2. One person can reserve up to 7 seats at a time.
3. If a person is reserving seats, the priority will be to book them in one row.
4. If seats are not available in one row then the booking should be done in such a way that the nearby seats are booked.
5. User can book as many tickets as s/he wants until the coach is full.
6. You don't have to create login functionality for this application.

How it should function?

1. Input required will only be the required number of seats. Example: 2 or 4 or 6 or 1 etc.
2. Output should be seats numbers that have been booked for the user along with the display of all the seats and their availability status through color or number or anything else that you may feel fit.

Submission Requirements:

1. Code (functions) as per the conditions and functionality mentioned above.
2. Database structure created as per the code.
3. Hosted on any platform, providing a working web URL.
4. Code submission in a zip file or Git link.

Solution Design:

Assumptions:

- Seats are numbered from 1 to 80.
- 11-row coach: 10 rows have 7 seats, last row has 3 seats.
- The user can book up to 7 seats at once, prioritizing booking in one row or nearby if not possible.
- Seat booking system ensures closest available seats are booked in case of split allocation.

Database Structure:

Table: 'Seats'

Column	Type	Description
seat_id	INT	Unique ID for each seat (Primary Key)
row_id	INT	Row number to which the seat belongs
status	VARCHAR(1)	'A' for available, 'B' for booked

Code Design:

Python + Flask web application:

1. Flask web server to handle HTTP requests.
2. SQLite database for storing seat data.
3. Functions to handle seat booking and seat allocation.
4. Simple HTML frontend to show seat status and allow bookings.

Sample Python Code:

```
from flask import Flask, request, jsonify, render_template

from flask_sqlalchemy import SQLAlchemy

app = Flask(__name__)

app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///seats.db'

db = SQLAlchemy(app)

class Seat(db.Model):

    seat_id = db.Column(db.Integer, primary_key=True)

    row_id = db.Column(db.Integer, nullable=False)

    status = db.Column(db.String(1), nullable=False, default='A')

with app.app_context():

    db.create_all()

def init_seats():

    with app.app_context():

        if Seat.query.count() == 0:

            for i in range(1, 81):

                row_id = (i - 1) // 7 + 1

                seat = Seat(seat_id=i, row_id=row_id)

                db.session.add(seat)

            db.session.commit()

init_seats()
```

```
def book_seats(seat_count):

    available_seats = Seat.query.filter_by(status='A').all()

    if len(available_seats) < seat_count:

        return None

    for row in range(1, 12):

        row_seats = Seat.query.filter_by(row_id=row, status='A').all()

        if len(row_seats) >= seat_count:

            booked_seats = row_seats[:seat_count]

            for seat in booked_seats:

                seat.status = 'B'

            db.session.commit()

            return booked_seats
```

```
booked_seats = available_seats[:seat_count]

for seat in booked_seats:

    seat.status = 'B'

db.session.commit()

return booked_seats
```

```
@app.route('/')
```

```
def home():

    seats = Seat.query.all()

    return render_template('seats.html', seats=seats)
```

```
@app.route('/book', methods=['POST'])
```

```
def book():

    seat_count = int(request.form['seat_count'])

    booked_seats = book_seats(seat_count)

    if booked_seats:

        seat_ids = [seat.seat_id for seat in booked_seats]

        return jsonify({'message': f"Seats booked: {seat_ids}", 'seats': seat_ids})

    else:

        return jsonify({'message': "Not enough seats available"})


if __name__ == '__main__':

    app.run(debug=True)
```

HTML Frontend (templates/seats.html):

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>Train Seat Reservation</title>

</head>

<body>

    <h1>Seat Reservation</h1>

    <table border="1">

        <tr>

            <th>Seat ID</th>

            <th>Row</th>
```

```
<th>Status</th>
```

```
</tr>
```

```
{% for seat in seats %}
```

```
<tr>
```

```
<td>{{ seat.seat_id }}</td>
```

```
<td>{{ seat.row_id }}</td>
```

```
<td>{{ 'Available' if seat.status == 'A' else 'Booked' }}</td>
```

```
</tr>
```

```
{% endfor %}
```

```
</table>
```

```
<h2>Book Seats</h2>
```

```
<form method="POST" action="/book">
```

```
<label for="seat_count">Enter number of seats:</label>
```

```
<input type="number" id="seat_count" name="seat_count" max="7" min="1" required>
```

```
<button type="submit">Book</button>
```

```
</form>
```

```
</body>
```

```
</html>
```