

Flight Booking and Management System

Introduction:

Description: This project is designed to manage flight bookings, including adding flights, booking tickets, confirming bookings, updating bookings, canceling bookings, and saving flight data to a JSON file. The system provides a user-friendly menu interface for interacting with the various functionalities.

Key Features

- Add new flights
- Display all available flights
- Book tickets for specific flights
- Confirm existing bookings
- Display all bookings
- Update existing bookings
- Cancel bookings
- Save flight data to a JSON file

System Requirements and Setup

System Requirements

Python Version: 3.x

Libraries: json

Setup Instructions

1. **Install Python:** Ensure Python 3.x is installed on your system.

2. **Install Required Libraries:** Use the following command to install the required libraries:

```
bash
```

```
pip install json
```

3. **Run the Application:** Execute the script using the following command:

```
bash
```

```
python airline_booking_system.py
```

Code Structure and Functions

Code Structure

Global Variables:

- `flights`: List to store flight objects.
- `bookings`: List to store booking objects.

Functions Overview

1. `book_ticket(booking_id, flight_number, seat_type, seat_position):`

- Books a ticket for a specified flight and seat.

2. `confirm_booking(booking_id):`

- Confirms a booking based on the booking ID.

3. `display_flights():`

- Displays all available flights.

4. `display_bookings():`

- Displays all bookings.

5. `update_booking(booking_id, new_seat_type, new_seat_position):`

- Updates an existing booking with new seat details.
- 6. `cancel_booking(booking_id)`:
 - Cancels a booking based on the booking ID.
- 7. `print_remaining_seats(flight_number)`:
 - Prints the remaining seats for a specified flight.
- 8. `write_flights_to_json(filename='flights_list.json')`:
 - Saves flight data to a JSON file.
- 9. `select_seat_type()`:
 - Prompts the user to select a seat type.
- 10. `select_seat_position()`:
 - Prompts the user to select a seat position.
- 11. `menu()`:
 - Displays the main menu and handles user input.
- 12. `menus()`:
 - Continuously displays the menu until the user chooses to exit.

Detailed Function Descriptions

`book_ticket(booking_id, flight_number, seat_type, seat_position)`

Purpose: Books a ticket for a specified flight and seat.

Parameters:

- `'booking_id'`: Unique identifier for the booking.
- `'flight_number'`: The flight number for which the ticket is being booked.
- `'seat_type'`: Type of seat (premium/business/regular).
- `'seat_position'`: Position of the seat (window/middle/aisle).

Returns: None

Example Usage:

```
```python
book_ticket(101, 'FL123', 'premium', 'window')
```
```

confirm_booking(booking_id)

Purpose: Confirms a booking based on the booking ID.

Parameters:

- 'booking_id': Unique identifier for the booking.

Returns: None

Example Usage:

```
```python
confirm_booking(101)
```
```

display_flights()

Purpose: Displays all available flights.

Parameters: None

Returns: None

Example Usage:

```
```python
display_flights()
```
```

display_bookings()

Purpose: Displays all bookings.

Parameters: None

Returns: None

Example Usage:

```
```python
display_bookings()
```
```

`update_booking(booking_id, new_seat_type, new_seat_position)`

Purpose: Updates an existing booking with new seat details.

Parameters:

- `'booking_id'`: Unique identifier for the booking.
- `'new_seat_type'`: New type of seat (premium/business/regular).
- `'new_seat_position'`: New position of the seat (window/middle/aisle).

Returns: None

Example Usage:

```
```python
update_booking(101, 'business', 'aisle')
```
```

User Interface and Interaction

Menu Interface

Description: The application provides a menu-driven interface for users to interact with the system.

Menu Options:

1. Add flight
2. Display all flights
3. Book ticket
4. Confirm booking

5. Display all bookings
6. Update booking
7. Cancel booking
8. Save flights to JSON
9. End

Example Interaction

- **Adding a Flight:**

```plaintext

Enter flight number: 1245

Enter flight name: Air India

Enter flight capacity: 300

```

Sample input:

```
1-Add flight
2-Display all flights
3-Book ticket
4-Confirm booking
5-Display all bookings
6-Update booking
7-Cancel booking
8-Save flights to JSON
9-End
Your choice: 1
Enter flight number: 1245
Enter flight name: Air India
Enter flight capacity: 300█
```

- ****Booking a Ticket:****

```plaintext

Enter booking ID: 1

Enter flight number: 123

Choose seat type (premium/business/regular): premium

Choose seat position (window/middle/aisle): window

```

```
Your choice: 3
Enter booking ID: 1
Enter flight number: 123
Choose seat type (premium/business/regular): premium
Choose seat position (window/middle/aisle): window
Ticket booked successfully
Remaining seats for flight 123: {'premium': {'window': 9, 'middle': 4, 'aisle': 6}, 'business': {'window': 20, 'middle': 10, 'aisle': 10}, 'regular': {'window': 70, 'middle': 40, 'aisle': 30}}
```

- ****Confirm Booking****

```plaintext

Enter booking ID: 1

Enter flight number: 123

Choose seat type (premium/business/regular): premium

Seat position: window

**\*\* Display All Bookings\*\***

It displays All the bookings as below mentioned picture

```
0-End
Your choice: 4
Enter booking ID: 2
Booking confirmed: [Booking ID=2, Flight Number=12, Seat Type=regular, Seat Position=aisle]
1-Add flight
2-Display all flights
3-Book ticket
4-Confirm booking
5-Display all bookings
6-Update booking
7-Cancel booking
8-Save flights to JSON
9-End
Your choice: 4
Enter booking ID: 1
Booking confirmed: [Booking ID=1, Flight Number=123, Seat Type=premium, Seat Position=window]
1-Add flight
2-Display all flights
3-Book ticket
4-Confirm booking
5-Display all bookings
6-Update booking
7-Cancel booking
8-Save flights to JSON
9-End
Your choice: 5
[Booking ID=1, Flight Number=123, Seat Type=premium, Seat Position=window]
[Booking ID=1, Flight Number=123, Seat Type=business, Seat Position=middle]
[Booking ID=2, Flight Number=12, Seat Type=regular, Seat Position=aisle]
1-Add flight
```