01

ONLINE FLIGHT BOOKING SYSTEM

SHAIRA TAHSIN ERA
NOSHIN NAWAL

Date: December 3, 2024

02

JAVA PROJECT

MADE WITH 4 SUB CLASSES

1. **BOOKING**
2. **FLIGHT MANAGER**
3. **FLIGHT**
4. **USER**

AND ONE MAIN CLASS: FLIGHT BOOKING APP

03

Flight Manager

```java
public class FlightManager {
    private static final int MAX_SEATS = 10;
    private int[][][] seats;

    String[] destinations = {"Chittagong", "Rajshahi", "Khulna", "Shylet", "Barishal"};
    String[] times = {"8am", "2pm", "8pm"};
    String[] dates = {"01/01/2025", "08/01/2025", "20/01/2025", "02/02/2025", "10/02/2025"};

    public FlightManager() {
        seats = new int[destinations.length][times.length][dates.length];
        initializeSeats();
    }

    private void initializeSeats() {
        for (int i = 0; i < destinations.length; i++) {
            for (int j = 0; j < times.length; j++) {
                for (int k = 0; k < dates.length; k++) {
                    seats[i][j][k] = MAX_SEATS;
                }
            }
        }
    }
```

- **MANAGES SEAT AVAILABILITY FOR FLIGHTS.**
- **TRACKS SEATS FOR EACH DESTINATION, DATE, AND TIME COMBINATION.**
- **USES A 3D ARRAY TO STORE AVAILABLE SEATS FOR ALL POSSIBLE FLIGHT COMBINATIONS.**

04

User

```java
public class User {
    private String username;
    private String password;
    private Booking[] bookings;
    private int bookingCount;

    public User(String username, String password) {
        this.username = username;
        this.password = password;
        this.bookings = new Booking[10];
        this.bookingCount = 0;
    }
```

- INCLUDES METHODS TO ADD AND RETRIEVE BOOKINGS.
- REPRESENTS A USER ACCOUNT, INCLUDING USERNAME, PASSWORD, AND BOOKINGS.
- STORES BOOKINGS IN AN ARRAY (UP TO 10 BOOKINGS).

UserManager

```java
public class UserManager {
    private static final String FILE_NAME = "users.txt";

    public UserManager() {
        File file = new File(pathname: FILE_NAME);
        if (!file.exists()) {
            try {
                file.createNewFile();
            } catch (IOException e) {
                System.out.println(x: "Error creating user data file.");
            }
        }
    }
```

- HANDLES USER DATA, INCLUDING LOGIN AND ACCOUNT CREATION.
- USES FILE I/O TO PERSIST USER DATA, INCLUDING THEIR BOOKINGS.
- VALIDATES USERNAME AND PASSWORD DURING LOGIN.

Flight Booking App

```java
public static void main(String[] args) {
    Scanner scanner = new Scanner(source: System.in);
    UserManager userManager = new UserManager();
    FlightManager flightManager = new FlightManager();

    while (true) {
        System.out.println(x: "1. Login");
        System.out.println(x: "2. Create Account");
        System.out.println(x: "3. Exit");
        System.out.print(s: "Enter choice: ");
        int choice = scanner.nextInt();
        scanner.nextLine();

        switch (choice) {
            case 1 -> loginUser(scanner, userManager, flightManager);
            case 2 -> createUser(scanner, userManager);
            case 3 -> {
                System.out.println(x: "Exiting...");
                scanner.close();
                return;
            }
            default -> System.out.println(x: "Invalid choice. Please try again.");
        }
    }
}
```

07

Output

```
1. Login
2. Create Account
3. Exit
Enter choice:
```

```
Login successful!
Main Menu:
1. Book a Flight
2. View Bookings
3. Logout
Enter choice:
```

- APPLICATION STARTUP
- MAIN MENU LOOP
- LOGIN PROCESS
- ACCOUNT CREATION
- MAIN MENU AFTER LOGIN
- FLIGHT BOOKING
- VIEWING BOOKINGS
- EXITING THE APPLICATION

08

Potential Problems

- LIMITED BOOKING CAPACITY
- HARDCODED FLIGHT DETAILS
- USERS ARE UNABLE TO VIEW AVAILABLE SEATS BEFORE BOOKING.

thank you