

Airline DataBase Management System Analysis

Introduction

The Airline Database Management System (ADMS) project aims to develop a comprehensive database solution for managing various aspects of airline operations, including bookings, flights, passengers, aircraft, and revenue analysis. The system will provide functionalities to efficiently handle booking transactions, flight scheduling, passenger information management, and revenue tracking.

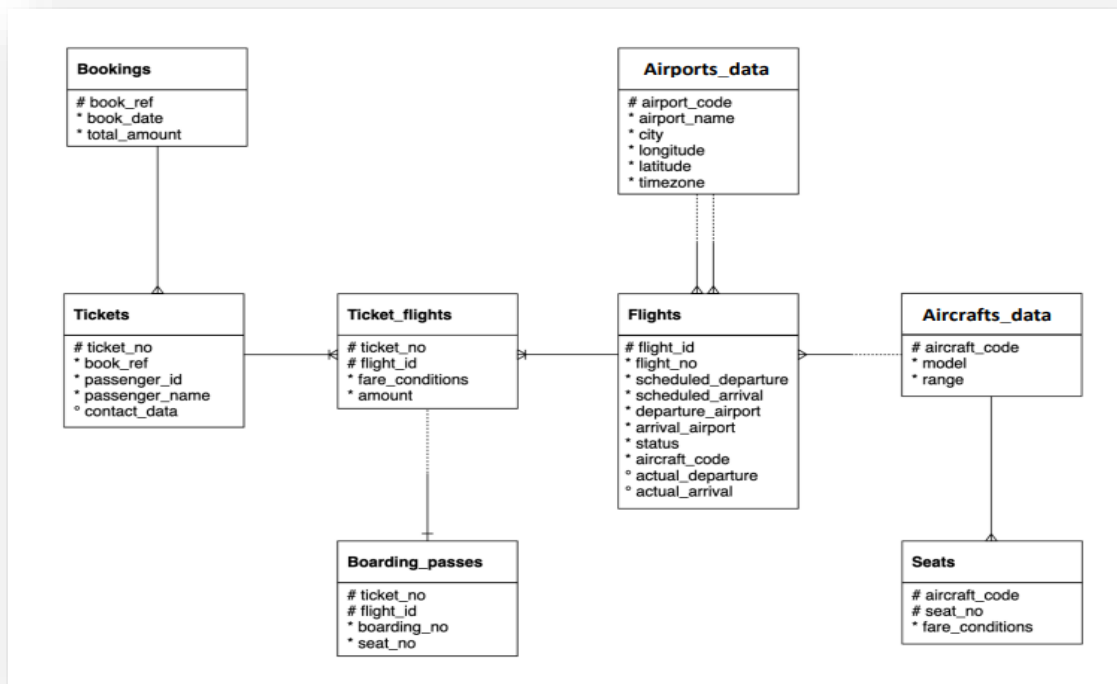
Project Objectives

- ✓ Design and implement a robust relational database schema to store airline-related data efficiently.
- ✓ Develop SQL queries to perform essential operations such as booking management, flight scheduling, and revenue analysis.
- ✓ Create a user-friendly interface to interact with the database system, enabling airline staff to perform tasks effectively.
- ✓ Ensure data integrity, security, and scalability of the database system.
- ✓ Provide detailed documentation for the project, including database schema, SQL queries, and user manual.

Database Schema

The database schema consists of the following entities:

- ✓ **Bookings:** Stores information about passenger bookings, including booking reference, booking date, and total amount.
- ✓ **Tickets:** Contains details of individual tickets issued to passengers, including ticket number, associated booking reference, passenger id, name and contact data.
- ✓ **Ticket_Flights:** Records the mapping between tickets and flight segments along with fare conditions.
- ✓ **Flights:** Stores flight information such as flight ID, departure airport, scheduled departure, and scheduled arrival times.
- ✓ **Aircrafts_data:** Contains details of aircraft code, model and cabin configuration.
- ✓ **Airport_data:** Records the airport name, city and code, along with latitude and longitude.
- ✓ **Boarding_passes:** Contains ticket, boarding and seat numbers
- ✓ **Seats:** Contains aircraft code, seat number and fare conditions



SQL Queries

1. Calculating the number of flights for each period of the day.

Problem Statement

Provide a SQL query that calculates the number of flights for each period of the day - morning, afternoon, evening, and night — with the time slots defined as follows. Flights departing before 12 PM(excluding) are 'Morning', those between 12 PM(including) and 3:59 PM are 'Afternoon', departures between 4 PM(including) and 7:59 PM are 'Evening', and any flights after 8 PM(including) are 'Night'. The query should output a table with five columns: the departure airport code, and the counts of morning, afternoon, evening, and night flights respectively. Please arrange the resulting table by departure airport code in ascending order. For extracting time, use scheduled departure time

Query

```

SELECT
    f.departure_airport,
    SUM(CASE WHEN EXTRACT(HOUR FROM f.scheduled_departure) < 12 THEN
1 ELSE 0 END) AS morning_flights,
    SUM(CASE WHEN EXTRACT(HOUR FROM f.scheduled_departure) >= 12 AND
EXTRACT(HOUR FROM f.scheduled_departure) < 16 THEN 1 ELSE 0 END) AS
afternoon_flights,

```

```

SUM(CASE WHEN EXTRACT(HOUR FROM f.scheduled_departure) >= 16 AND
EXTRACT(HOUR FROM f.scheduled_departure) < 20 THEN 1 ELSE 0 END) AS
evening_flights,
SUM(CASE WHEN EXTRACT(HOUR FROM f.scheduled_departure) >= 20
THEN 1 ELSE 0 END) AS night_flights
FROM
    flights f
GROUP BY
    f.departure_airport
ORDER BY
    f.departure_airport;

```

Output:

OUTPUT	FLIGHTS	BOARDING_PASSES	AIRCRAFTS	TICKET_FLIGHTS	SEATS	BOOKINGS	TICKETS	AIRPORTS
departure_airport	morning_flights		afternoon_flights		evening_flights		night_flights	
AAQ	1		125		0		0	
ABA	23		57		0		0	
AER	10		38		221		149	
ARH	0		78		55		0	
ASF	0		74		0		0	
BAX	40		15		16		0	
BQS	0		26		0		0	
BTK	47		0		0		0	
BZK	40		20		135		88	
CEE	3		23		0		0	
CEK	29		63		58		47	
CNN	14		15		0		0	
Showing 104 records								

2. Determining the contribution of monthly revenue to the overall yearly revenue.

Problem Statement

The airline wants to calculate the contribution of monthly revenue generated by bookings to the overall revenue of the year. Create a table having year, month and revenue contribution (up to 2 decimal points, it should be less than 1, no need to convert it to percentage). Sort the table in ascending order of year(YYYY) followed by ascending order of month(MM)

Query

```

SELECT
    EXTRACT(YEAR FROM b.book_date) AS year,
    EXTRACT(MONTH FROM b.book_date) AS month,
    ROUND(SUM(b.total_amount) / yearly_revenue.total_revenue, 2) AS
revenue_contribution
FROM
    bookings b
JOIN (

```

```

SELECT
    EXTRACT(YEAR FROM book_date) AS year,
    SUM(total_amount) AS total_revenue
FROM
    bookings
GROUP BY
    EXTRACT(YEAR FROM book_date)
) AS yearly_revenue ON EXTRACT(YEAR FROM b.book_date) =
yearly_revenue.year
GROUP BY
    EXTRACT(YEAR FROM b.book_date),
    EXTRACT(MONTH FROM b.book_date),
    yearly_revenue.total_revenue
ORDER BY
    year ASC,
    month ASC;

```

Output:

OUTPUT	FLIGHTS	BOARDING_PASSES	AIRCRAFTS	TICKET_FLIGHTS	SEATS	BOOKINGS	TICKETS	AIRPORTS
year	month						revenue_contribution	
2016	7						0.00	
2016	8						0.00	
2016	9						0.00	
2016	10						0.00	
2016	11						0.00	
2016	12						0.00	
2017	1						0.00	
2017	2						0.00	
2017	3						0.00	
2017	4						0.00	
2017	5						0.00	
2017	6						0.00	
Showing 14 records								

3. Identifying booking references where travellers took Airbus flights.

Problem Statement

Find out all the booking references where travelers took 'Airbus' Flights. Expected output is list of all the booking references sorted in ascending order

Query

```

SELECT DISTINCT
    b.book_ref
FROM
    bookings b
JOIN

```

```

        tickets t ON b.book_ref = t.book_ref
JOIN
        ticket_flights tf ON t.ticket_no = tf.ticket_no
JOIN
        flights f ON tf.flight_id = f.flight_id
JOIN
        aircrafts a ON f.aircraft_code = a.aircraft_code
WHERE
        TRIM(a.model) LIKE '%Airbus%'
ORDER BY
        b.book_ref ASC

```

Output:

OUTPUT	FLIGHTS	BOARDING_PASSES	AIRCRAFTS	TICKET_FLIGHTS	SEATS	BOOKINGS	TICKETS	AIRPORTS
book_ref								
002BCF								
0068A1								
0077CA								
010415								
013C3C								
017D9A								
01B689								
032CE9								
038328								
03BF7E								
03EF37								
04548E								
Showing 500 records								

4. Finding total scheduled flying time for bookings with more than one flight.

Problem Statement

Identify total SCHEDULED flying time for all booking references having more than 1 flight. Expected output book_ref and total flying time

Query

```

WITH BookingFlights AS (
    SELECT
        b.book_ref,
        COUNT(tf.flight_id) AS num_flights,
        SUM(EXTRACT(EPOCH FROM (fl.scheduled_arrival -
fl.scheduled_departure))) AS total_flying_time
    FROM
        bookings b
    JOIN
        tickets t ON b.book_ref = t.book_ref

```

```

JOIN
    ticket_flights tf ON t.ticket_no = tf.ticket_no
JOIN
    flights fl ON tf.flight_id = fl.flight_id
GROUP BY
    b.book_ref
HAVING
    COUNT(tf.flight_id) > 1
)
SELECT
    book_ref,
    total_flying_time
FROM
    BookingFlights;

```

Output:

OUTPUT	FLIGHTS	BOARDING_PASSES	AIRCRAFTS	TICKET_FLIGHTS	SEATS	BOOKINGS	TICKETS	AIRPORTS
book_ref	total_flying_time							
8C815D	126000							
34BC05	24600							
1E42A6	40800							
9B99A7	12600							
2C1044	16800							
DD9A76	59400							
0A6973	35400							
5D6A21	25800							
7EF764	61200							
AF94D3	25200							
F719CF	59400							
E801FF	64200							
Showing 500 records								

Airline Database: [LINK](#)

Conclusion

The Airline Database Management System project aims to streamline airline operations by providing a centralized platform for managing bookings, flights, passengers, and revenue. With its robust database schema, efficient SQL queries, and user-friendly interface, the system will enhance the efficiency and effectiveness of airline management processes.

BY – DIVYA GATTUGARI