Designing a Relational Database

The Conceptual Schema:

Before defining the logical schema of the database, we should clarify following information:

- Can the hotel name be used as a primary key?
    - No, Hotel names are not guaranteed to be unique. For example, multiple hotels across different cities or countries might share the same name (e.g., “Grand Hotel” or “Hilton”).
        - ➢ Use a **HotelID** (a unique, system-generated identifier) as the primary key to ensure uniqueness and avoid conflicts.
- Can a flight number be used as a primary key to identify a flight?
    - No, The same airline can operate the same flight number on different dates (e.g., Flight BA123 operates daily).
        - ➢ Define a unique **FlightID** for each specific flight instance or a **composite key** consisting of **FlightNumber, AirlineID, FlightDate**.
- Even if it is unlikely that 2 reviews have the same textual content, should review content be used as a primary key?
    - No, text fields are variable and potentially large, leading to performance and storage issues. Primary keys must be short, fixed-length, and efficient for indexing.
        - ➢ Use a **ReviewID** as the primary key.
- IATA code uniquely identifies an airport, should it be used as a primary key for the entity **Airport**?
    - Yes, possible as not all airports may have an IATA code. However, it is better to use a system-generated **AirportID** as the primary key.

1. Entity and Relation

Based on the application context and previous clarification, we can summarize Entities with Primary Keys:

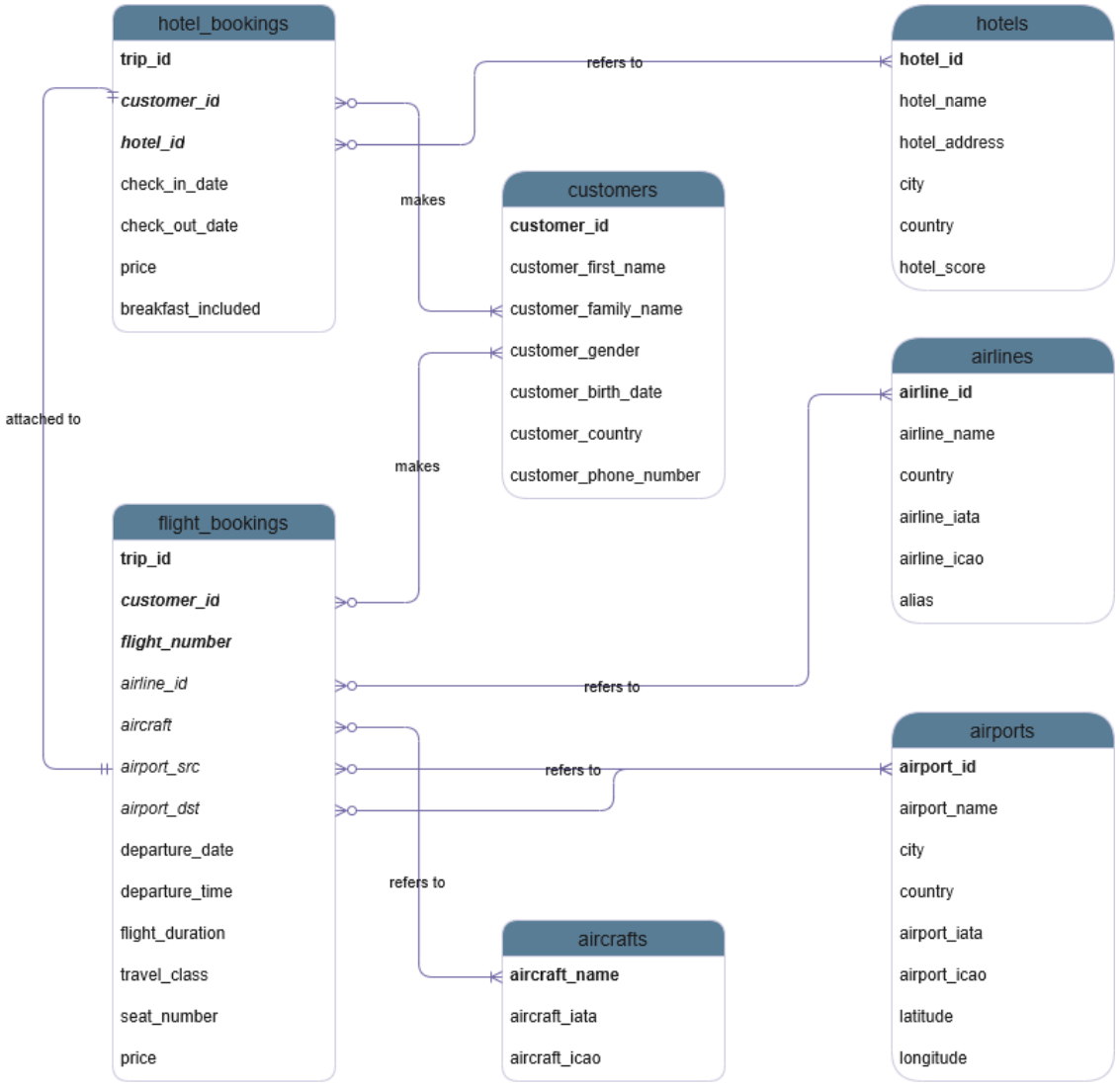
- **Customer**: *CustomerID*, FirstName, LastName, Gender, DateOfBirth, Phone, Country.
- **Airline**: *AirlineID*, Name, Country, IATA, ICAO, Alias.
- **Airport**: *AirportID*, Name, City, Country, Latitude, Longitude, IATA, ICAO.
- **Aircraft**: *IATA, ICAO*, Name.

- **Hotel**: *HotelID*, Name, Address, City, Country, AvgReviewScore.
- **FlightBooking**: *TripID*, CustomerID, FlightNumber, AirlineID, Aircraft, AirportSrc, AirportDst, DepartureTime, DepartureDate, FlightDuration, TravelClass, SearNumber, Price
- **HotelBooking**: *TripID*, HotelID, CustomerID, CheckInDate, CheckOutDate, Price, BreakfastIncluded

Relationships:

- A **Customer** can make many **FlightBookings** and **HotelBookings**.
- Each **FlightBooking** is linked to one **Airline**, **Aircraft**, and two **Airports**.
- Each **HotelBooking** is linked to one **Hotel**.

We can create a **ERD (Entity Relationship Diagram)** to have an initial view of the final database:



2. Tables from Conceptual Schema

Table Name	Entity/Relation	Primary Key	Other Columns
Customer	Customer	CustomerID	FirstName, LastName, Gender, DateOfBirth, Phone, Country
Airline	Airline	AirlineID	Name, Country, IATA, ICAO, Alias
Airport	Airport	AirportID	Name, City, Country, Latitude, Longitude, IATA, ICAO.
Aircraft	Aircraft	IATA, ICAO	AircraftName
Hotel	Hotel	HotelID	Name, Address, City, Country, AvgReviewScore.
FlightBooking	FlightBooking	TripID, CustomerID, FlightNumber	*AirlineID*, *Aircraft*, *AirportSrc*, *AirportDst*, DepartureTime, DepartureDate, FlightDuration, TravelClass, SearNumber, Price
HotelBooking	HotelBooking	TripID, CustomerID, HotelID	CheckInDate, CheckOutDate, Price, BreakfastIncluded

3. Functional Dependencies and Normal Forms

Currently, all non-key attributes depend fully on the primary key, and there are no transitive dependencies. Therefore, all tables are in 3NF now.

4. Normalizing Tables to 3NF

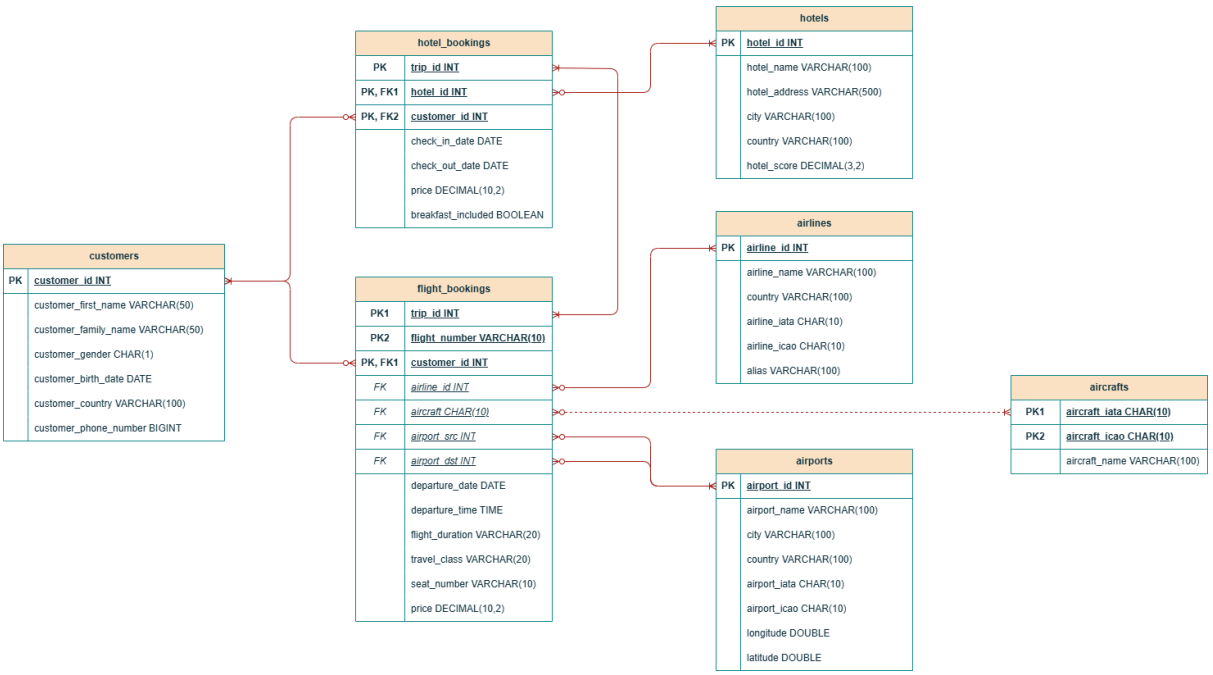
From what we have modelled, most tables are already in 3NF due to:

- Use of synthetics IDs.
- No partial or transitive dependencies.
- Atomic attributes.

Therefore, normalization is not required at this stage.

5. Designing Physical Schema

By defining all tables, attributes, relationships; a complete physical schema is created:


hotels
PK
hotel_id INT
hotel_name VARCHAR(100)
hotel_address VARCHAR(500)
city VARCHAR(100)
country VARCHAR(100)
hotel_score DECIMAL(3,2)
hotel_bookings
PK
trip_id INT
PK, FK1
hotel_id INT
PK, FK2
customer_id INT
check_in_date DATE
check_out_date DATE
price DECIMAL(10,2)
breakfast_included BOOLEAN
airlines
PK
airline_id INT
airline_name VARCHAR(100)
country VARCHAR(100)
airline_iata CHAR(10)
airline_icao CHAR(10)
alias VARCHAR(100)
customers
PK
customer_id INT
customer_first_name VARCHAR(50)
customer_family_name VARCHAR(50)
customer_gender CHAR(1)
customer_birth_date DATE
customer_country VARCHAR(100)
customer_phone_number BIGINT
flight_bookings
PK1
trip_id INT
PK2
flight_number VARCHAR(10)
PK, FK1
customer_id INT
FK
airline_id INT
FK
aircraft CHAR(10)
FK
airport_src INT
FK
airport_dst INT
departure_date DATE
departure_time TIME
flight_duration VARCHAR(20)
travel_class VARCHAR(20)
seat_number VARCHAR(10)
price DECIMAL(10,2)
aircrafts
PK1
aircraft_iata CHAR(10)
PK2
aircraft_icao CHAR(10)
aircraft_name VARCHAR(100)
airports
PK
airport_id INT
airport_name VARCHAR(100)
city VARCHAR(100)
country VARCHAR(100)
airport_iata CHAR(10)
airport_icao CHAR(10)
longitude DOUBLE
latitude DOUBLE