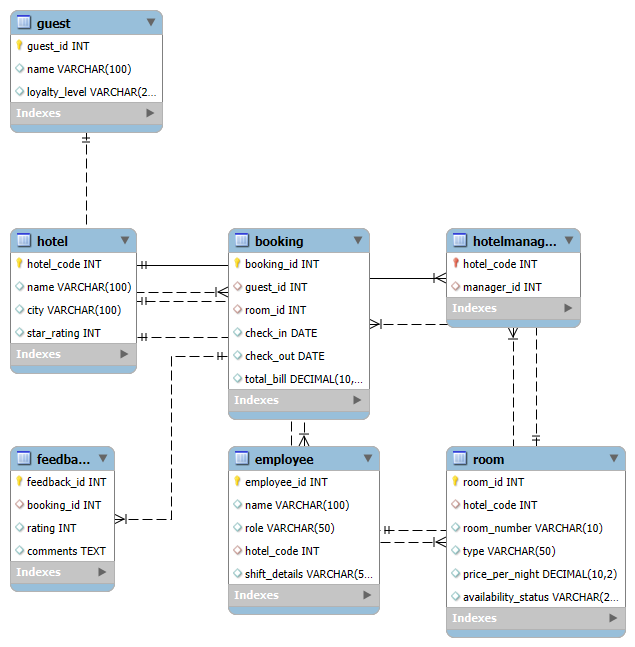
**Problem: 4 Multi-city Hotel Chain Management System**

Design an Entity-Relationship schema for a multi-city hotel chain management system. The database must maintain hotels identified by hotel code, name, city, manager, number of rooms, and star rating. Rooms have room number, type, price per night, availability status, and belong to a hotel.

Guests have guest ID, name, loyalty level, booking history, and feedback given for bookings. Bookings have booking ID, guest, room, check-in and check-out dates, and total bill. Employees have employee ID, name, role, hotel assigned, and shift details.

Each hotel has multiple rooms and employees, and is managed by a manager who is also an employee. Guests can book rooms in any hotel and can have multiple active or past bookings. Rooms can be booked by different guests over time, but only one guest can occupy a room at a given time.

Employees are assigned to a specific hotel and can work in different shifts and roles. Loyalty level of a guest is updated based on their booking history and feedback. Feedback is linked to specific bookings and can influence loyalty level updates.



**-- Hotel table**

CREATE TABLE Hotel (

hotel\_code INT PRIMARY KEY,

name VARCHAR(100),

city VARCHAR(100),

star\_rating INT

);

**-- Room table**

CREATE TABLE Room (

room\_id INT PRIMARY KEY AUTO\_INCREMENT,

hotel\_code INT,

room\_number VARCHAR(10),

type VARCHAR(50),

price\_per\_night DECIMAL(10,2),

availability\_status VARCHAR(20),

FOREIGN KEY (hotel\_code) REFERENCES Hotel(hotel\_code)

);

**-- Guest table**

CREATE TABLE Guest (

guest\_id INT PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(100),

loyalty\_level VARCHAR(20)

);

**-- Booking table**

CREATE TABLE Booking (

booking\_id INT PRIMARY KEY AUTO\_INCREMENT,

guest\_id INT,

room\_id INT,

check\_in DATE,

check\_out DATE,

total\_bill DECIMAL(10,2),

FOREIGN KEY (guest\_id) REFERENCES Guest(guest\_id),

FOREIGN KEY (room\_id) REFERENCES Room(room\_id)

);

**-- Feedback table**

CREATE TABLE Feedback (

feedback\_id INT PRIMARY KEY AUTO\_INCREMENT,

booking\_id INT,

rating INT CHECK (rating BETWEEN 1 AND 5),

comments TEXT,

FOREIGN KEY (booking\_id) REFERENCES Booking(booking\_id)

);

**-- Employee table**

CREATE TABLE Employee (

employee\_id INT PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(100),

role VARCHAR(50),

hotel\_code INT,

shift\_details VARCHAR(50),

FOREIGN KEY (hotel\_code) REFERENCES Hotel(hotel\_code)

);

**-- Manager Assignment (1 manager per hotel, and manager is an employee)**

CREATE TABLE HotelManager (

hotel\_code INT PRIMARY KEY,

manager\_id INT,

FOREIGN KEY (hotel\_code) REFERENCES Hotel(hotel\_code),

FOREIGN KEY (manager\_id) REFERENCES Employee(employee\_id)

);