

SQL Exercises

**This material comes from the
recommended book by
T. Connolly, C. Begg, A. Strachan**

The Database

Hotel (Hotel_No, Name, Address)

Room (Room_No, Hotel_No, Type, Price)

Booking (Hotel_No, Guest_No, Date_From,
Date_To, Room_No)

Guest (Guest_No, Name, Address)

Creating the Tables

(Ignoring PKs and FKs)

```
CREATE TABLE hotel  
( hotel_no CHAR(4) NOT NULL,  
  name VARCHAR(20) NOT NULL,  
  address VARCHAR(50) NOT NULL);
```

```
CREATE TABLE room  
( room_no VARCHAR(4) NOT NULL,  
  hotel_no CHAR(4) NOT NULL,  
  type CHAR(1) NOT NULL,  
  price DECIMAL(5,2) NOT NULL);
```

```
CREATE TABLE booking
(hotel_no CHAR(4) NOT NULL,
guest_no CHAR(4) NOT NULL,
date_from DATETIME NOT NULL,
date_to DATETIME NULL,
room_no CHAR(4) NOT NULL);
```

Dates: YYYY-MM-DD

```
CREATE TABLE guest  
( guest_no CHAR(4) NOT NULL,  
  name VARCHAR(20) NOT NULL,  
  address VARCHAR(50) NOT NULL);
```

Populating the Tables

```
INSERT INTO hotel
VALUES ('H111', 'Grosvenor Hotel', 'London');
INSERT INTO room
VALUES ('1', 'H111', 'S', 72.00);
INSERT INTO guest
VALUES ('G111', 'John Smith', 'London');
INSERT INTO booking
VALUES ('H111', 'G111', DATE'1999-01-01',
      DATE'1999-01-02', '1');
```

Updating the Tables

```
UPDATE room SET price = price*1.05;
```

- Create a separate table with the same structure as the Booking table to hold archive records.
- Using the INSERT statement, copy the records from the Booking table to the archive table relating to bookings before 1st January 2000. Delete all bookings before 1st January 2000 from the Booking table.

```
CREATE TABLE booking_old  
( hotel_no CHAR(4) NOT NULL,  
  guest_no CHAR(4) NOT NULL,  
  date_from DATETIME NOT NULL,  
  date_to DATETIME NULL,  
  room_no VARCHAR(4) NOT NULL);
```

```
INSERT INTO booking_old  
(SELECT * FROM booking  
WHERE date_to < DATE'2000-01-01');  
DELETE FROM booking  
WHERE date_to < DATE'2000-01-01';
```

Queries: Back to the Database

Hotel (Hotel_No, Name, Address)

Room (Room_No, Hotel_No, Type, Price)

Booking (Hotel_No, Guest_No, Date_From,
Date_To, Room_No)

Guest (Guest_No, Name, Address)

Simple Queries

1. List full details of all hotels.
2. List full details of all hotels in London.
3. List the names and addresses of all guests in London, alphabetically ordered by name.
4. List all double or family rooms with a price below £40.00 per night, in ascending order of price.
5. List the bookings for which no date_to has been specified.

Aggregate Functions

1. How many hotels are there?
2. What is the average price of a room?
3. What is the total revenue per night from all double rooms?
4. How many different guests have made bookings for August?

Subqueries and Joins

1. List the price and type of all rooms at the Grosvenor Hotel.
2. List all guests currently staying at the Grosvenor Hotel.
3. List the details of all rooms at the Grosvenor Hotel, including the name of the guest staying in the room, if the room is occupied.
4. What is the total income from bookings for the Grosvenor Hotel today?
5. List the rooms that are currently unoccupied at the Grosvenor Hotel.
6. What is the lost income from unoccupied rooms at the Grosvenor Hotel?

Grouping

1. List the number of rooms in each hotel.
2. List the number of rooms in each hotel in London.
3. What is the average number of bookings for each hotel in August?
4. What is the most commonly booked room type for each hotel in London?
5. What is the lost income from unoccupied rooms at each hotel today?