Mohamed KHIDER BISKRA University
Faculty of Exact Sciences
Department of computer science
Level: 2 LMD
Module: Database

Lab Work n° 2

In the hospitality sector, effective reservation management is crucial to ensure optimal room occupancy and provide quality service to customers. Hotel establishments must manage several aspects, such as:
• **Hotel Information**: Each hotel has unique characteristics (name, address, contact details) that must be accurately recorded.
• **Room Management**: Rooms vary by type (single, double, suite, etc.), pricing, and amenities. Each room is associated with a specific hotel.
• **Client Management**: To make a reservation, it is necessary to collect clients' personal information (first name, last name, email, phone number, etc.).
• **Reservation Management**: Each reservation involves linking a client to a room for a specific period and must adhere to time constraints (start date and end date).

Objectives

• Design and implement a relational database to model key entities (Hotel, Room, Client, Reservation) and their relationships.
• Apply integrity constraints to ensure data consistency and reliability.
• Implement control mechanisms (CHECK constraints) to validate data entries and prevent errors during insertion or update operations.

Consider the database described by the following tables:

```
Hotel (id, name, address, phone);
Room (id_r, number, type, price, #id);
Customer (id_cu, last_name, first_name, email, phone);
Reservation (#id_cu, #id_r, start_date, end_date);
```

Table Descriptions

- **Hotel:** Contains the hotel's identifier (id), name, address, and phone number.
- **Room:** Each room has a unique identifier (id_r), a room number, type, price (as a decimal value), and a foreign key (id) linking it to the corresponding hotel.
    - The foreign key constraint fk_room_hotel ensures the room is associated with an existing hotel.
- **Customer:** Stores basic customer information, including an identifier (id_cu), last name, first name, email, and phone number.
- **Reservation:** Links a customer to a room for a specific period defined by start_date and end_date.
    - The columns id_cu and id_r are foreign keys referencing the Customer and Room tables, respectively.

Note: The # symbol in the table definitions likely denotes foreign keys (e.g., #id refers to the id column in the Hotel table). The translation retains the original structure and naming conventions for clarity.

Questions

Create Database

1. Create a database named HotelReservationGx such that x is the number of the group
2. Create the tables:

```
Hotel (id, name, address, phone);      (Use AUTO_INCREMENT for id)
Room (id_r, number, type, price, #id);
Customer (id_cu, last_name, first_name, email, phone);
Reservation (#id_cu, #id_r, start_date, end_date);
```

3. Add some integrity constraints:

- Add the non-null constraint to the necessary fields(attributes);
- Add the DEFAULT="DOUBLE" constraint to the TYPE attribute;
- add the UNIQUE constraint to the Email attribute;
- Constraint on the room price: Ensure that the room price is strictly positive;
- Constraint on the reservation period: Verify that the start date is earlier than the end date.

Data Insertion

• Insert into the Hotel table:
('Hotel Paradis', '123 Rue du Soleil, Paris', '0123456789'),
('Hotel des Etoiles', '45 Avenue de la Lune, Lyon', '0987654321');
• Insert into the Room table:
(1,'101', 'Single', 80.00, 1),
(2,'102', 'Double', 120.00, 1),
(3,'201', 'Suite', 200.00, 2),
(4,'202', 'Double', 150.00, 2);
• Insert into the Customer table:
(1,'Durand', 'Alice', 'alice.durand@example.com', '0601020304'),
(2,'Lefevre', 'Bob', 'bob.lefevre@example.com', '0708091011');
• Insert into the Reservation table:
(1, 1, '2025-06-01', '2025-06-05'),
(2, 3, '2025-07-10', '2025-07-15');

Schema Modification

• Adding a column:
Add a column “description” to the Room table to describe the room’s amenities.
• Modifying the data type:
Increase the size of the email field in the Customer table to accommodate longer addresses.
• Dropping a column:
If you decide to no longer use the phone column in the Hotel table, you can drop it.

Questions on the Inserted Data

Question 1: List of Hotels;
Question 2: List of Customers;
Question 3: List of Reservations by Customer;
Question 4: Modify the value of the Type attribute "from DOUBLE to SUITE" of the ROOM table whose id_r=4.
Question 5: Display, for each customer who has made at least one reservation:
• The customer’s last name and first name
• The total number of reservations made by that customer

Question 6: Display all reservations whose start date is on or after July 1, 2025.
For each reservation, display the customer’s name, room number, hotel name, start date, and end date.

Exercises on SQL Joins

Question 1: Simple Join (INNER JOIN)
Display for each reservation the following information:
• The customer’s last name and first name
• The room number and type
• The hotel’s name
• The reservation’s start date and end date

Question 2: Left Join
Display the list of all customers. For each customer, display their reservation details (start date, end date, room number, and hotel name) if they have one. If a customer does not have a reservation, display NULL values for these details.

Question 3: Join to Display All Rooms
Display all rooms of each hotel, even if they are not reserved. For each room, display the room number, type, hotel name and, if the room is reserved, the reservation dates.

Question 4: Join with Aggregation
Display the total number of reservations per hotel. To do this, join the Hotel, Room, and Reservation tables and group the result by hotel name.

Question 5: Calculation of Stay Duration
Display, for each reservation, the customer’s name, room number, hotel name, start and end dates, as well as the duration of the stay in days (using the DATEDIFF function).

Good Luck