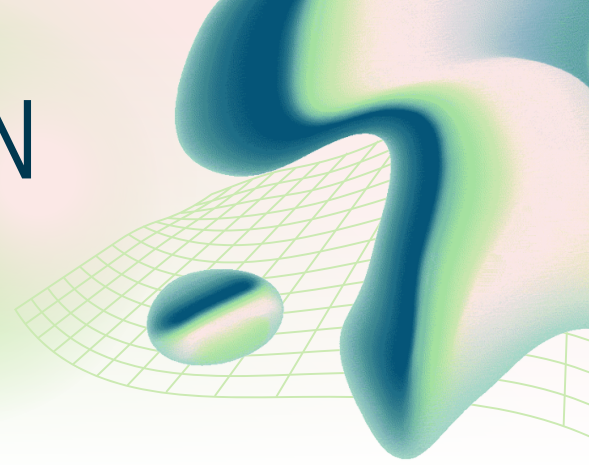


# SYSTÈME DE GESTION HÔTELIÈRE



**Binôme :**

Nom complet.	Numéro d'apogée
ELMOUHIB Ikram	2125072
EL HASOUBI Soukaina	213126

**Enseignant :** Pr. JIHAD ZAHIR

**Formation :** Licence MIP S4

**Faculté :** Faculté des Sciences Semlalia – Université Cadi Ayyad

**Année universitaire :** 2024 – 2025

---

 **Vidéo de démonstration du projet :**

[https://drive.google.com/drive/folders/1WZZG6t2kszJGJSk7uSboSzfWaBSISMhh?usp=drive\\_link](https://drive.google.com/drive/folders/1WZZG6t2kszJGJSk7uSboSzfWaBSISMhh?usp=drive_link)

---

 **Repository GitHub du projet :**

<https://github.com/ikrammouhib/Projet-BD>

---

# 1. Création de la base de données et des tables

```
DROP DATABASE IF EXISTS gestion_hoteliere;
```

```
CREATE DATABASE gestion_hoteliere;
```

```
USE gestion_hoteliere;
```

```
-- Table Hôtel
```

```
CREATE TABLE Hotel (  
id_hotel INT PRIMARY KEY,  
ville VARCHAR(100) NOT NULL,  
pays VARCHAR(100) NOT NULL,  
code_postal VARCHAR(10)  
);
```

```
-- Table Client
```

```
CREATE TABLE Client (  
id_client INT PRIMARY KEY,  
adresse VARCHAR(255),  
ville VARCHAR(100),  
code_postal VARCHAR(10),  
email VARCHAR(150),  
telephone VARCHAR(20),  
nom_complet VARCHAR(100) NOT NULL  
);
```

```
-- Table Prestation
```

```
CREATE TABLE Prestation (  
id_prestation INT PRIMARY KEY,  
prix DECIMAL(10,2),  
nom_prestation VARCHAR(100) NOT NULL  
);
```

```
-- Table TypeChambre
```

```
CREATE TABLE TypeChambre (  
id_type INT PRIMARY KEY,  
type_nom VARCHAR(50) NOT NULL,  
prix_nuit DECIMAL(10,2) NOT NULL  
);
```

```
-- Table Chambre
```

```
CREATE TABLE Chambre (  
.....
```

```
id_chambre INT PRIMARY KEY,  
numero_chambre INT NOT NULL,  
etage INT,  
balcon BOOLEAN DEFAULT FALSE,  
id_hotel INT,  
id_type INT,  
FOREIGN KEY (id_hotel) REFERENCES Hotel(id_hotel),  
FOREIGN KEY (id_type) REFERENCES TypeChambre(id_type)  
);
```

-- Table Réservation

```
CREATE TABLE Reservation (  
id_reservation INT PRIMARY KEY,  
date_arrivee DATE NOT NULL,  
date_depart DATE NOT NULL,  
id_client INT,  
FOREIGN KEY (id_client) REFERENCES Client(id_client)  
);
```

-- Table Evaluation

```
CREATE TABLE Evaluation (  
id_evaluation INT PRIMARY KEY,  
date_evaluation DATE,  
note INT CHECK (note >= 1 AND note <= 5),  
commentaire TEXT,  
id_reservation INT,  
FOREIGN KEY (id_reservation) REFERENCES Reservation(id_reservation)  
);
```

-- Table de liaison Réservation\_Chambre (relation N:M)

```
CREATE TABLE ReservationChambre (  
id_reservation INT,  
id_chambre INT,  
PRIMARY KEY (id_reservation, id_chambre),  
FOREIGN KEY (id_reservation) REFERENCES Reservation(id_reservation),  
FOREIGN KEY (id_chambre) REFERENCES Chambre(id_chambre)  
);
```

-- Table de liaison Client\_Prestation (relation N:M)

```
CREATE TABLE ClientPrestation (  
id_client INT,
```

```
id_prestation INT,  
PRIMARY KEY (id_client, id_prestation),  
FOREIGN KEY (id_client) REFERENCES Client(id_client),  
FOREIGN KEY (id_prestation) REFERENCES Prestation(id_prestation)  
);
```

---

## 2. Insertion des données

-- Données Hôtel

```
INSERT INTO Hotel VALUES
```

```
(1, 'Paris', 'France', '75001'),
```

```
(2, 'Lyon', 'France', '69002');
```

-- Données Client

```
INSERT INTO Client VALUES
```

```
(1, '12 Rue de Paris', 'Paris', '75001', 'jean.dupont@email.fr', '0612345678', 'Jean Dupont'),
```

```
(2, '5 Avenue Victor Hugo', 'Lyon', '69002', 'marie.leroy@email.fr', '0623456789', 'Marie Leroy'),
```

```
(3, '8 Boulevard Saint-Michel', 'Marseille', '13005', 'paul.moreau@email.fr', '0634567890', 'Paul Moreau'),
```

```
(4, '27 Rue Nationale', 'Lille', '59800', 'lucie.martin@email.fr', '0645678901', 'Lucie Martin'),
```

```
(5, '3 Rue des Fleurs', 'Nice', '06000', 'emma.giraud@email.fr', '0656789012', 'Emma Giraud');
```

-- Données Prestation

```
INSERT INTO Prestation VALUES
```

```
(1, 15.00, 'Petit-déjeuner'),
```

```
(2, 30.00, 'Navette aéroport'),
```

```
(3, 0.00, 'Wi-Fi gratuit'),
```

```
(4, 50.00, 'Spa et bien-être'),
```

```
(5, 20.00, 'Parking sécurisé');
```

-- Données TypeChambre

```
INSERT INTO TypeChambre VALUES
```

```
(1, 'Chambre Simple', 80.00),
```

```
(2, 'Chambre Double', 120.00);
```

-- Données Chambre

**INSERT INTO Chambre VALUES**

(1, 201, 2, FALSE, 1, 1),

(2, 502, 5, TRUE, 1, 2),

(3, 305, 3, FALSE, 2, 1),

(4, 410, 4, FALSE, 2, 2),

(5, 104, 1, TRUE, 2, 2),

(6, 202, 2, FALSE, 1, 1),

(7, 307, 3, TRUE, 1, 2),

(8, 101, 1, FALSE, 1, 1);

**-- Données Réservation**

**INSERT INTO Reservation VALUES**

(1, '2025-06-15', '2025-06-18', 1),

(2, '2025-07-01', '2025-07-05', 2),

(3, '2025-08-10', '2025-08-14', 3),

(4, '2025-09-05', '2025-09-07', 4),

(5, '2025-09-20', '2025-09-25', 5),

(7, '2025-11-12', '2025-11-14', 2),

(9, '2026-01-15', '2026-01-18', 4),

(10, '2026-02-01', '2026-02-05', 2);

**-- Liaison Réservation-Chambre**

**INSERT INTO ReservationChambre VALUES**

(1, 1), (2, 2), (3, 3), (4, 4), (5, 5),

(7, 7), (9, 4), (10, 2);

**-- Données Evaluation**

**INSERT INTO Evaluation VALUES**

(1, '2025-06-15', 5, 'Excellent séjour, personnel très accueillant.', 1),

(2, '2025-07-01', 4, 'Chambre propre, bon rapport qualité/prix.', 2),

(3, '2025-08-10', 3, 'Séjour correct mais bruyant la nuit.', 3),

(4, '2025-09-05', 5, 'Service impeccable, je recommande.', 4),

(5, '2025-09-20', 4, 'Très bon petit-déjeuner, hôtel bien situé.', 5);

---

### 3. Requêtes SQL demandées

- **Afficher la liste des réservations avec le nom du client et la ville de l'hôtel réservé**

```
SELECT
    r.id_reservation,
    c.nom_complet AS nom_client,
    h.ville AS ville_hotel,
    r.date_arrivee,
    r.date_depart
FROM Reservation r
JOIN Client c ON r.id_client = c.id_client
JOIN ReservationChambre rc ON r.id_reservation = rc.id_reservation
JOIN Chambre ch ON rc.id_chambre = ch.id_chambre
JOIN Hotel h ON ch.id_hotel = h.id_hotel;
```

Cette requête récupère toutes les réservations, en joignant les tables pour obtenir le nom complet du client et la ville où se trouve l'hôtel réservé.

- 
- **Afficher les clients qui habitent à Paris**

```
SELECT * FROM Client
WHERE ville = 'Paris';
```

Cette requête sélectionne tous les clients dont la ville est « Paris ».

- 
- **Calculer le nombre de réservations faites par chaque client**

```
SELECT
    c.nom_complet,
    COUNT(r.id_reservation) AS nombre_reservations
FROM Client c
LEFT JOIN Reservation r ON c.id_client = r.id_client
GROUP BY c.id_client, c.nom_complet
ORDER BY nombre_reservations DESC;
```

Cette requête compte le nombre total de réservations effectuées par chaque client, en affichant également leur nom. On utilise un LEFTJOIN pour inclure aussi les clients sans réservation

- 
- **Donner le nombre de chambres pour chaque type de chambre**

```
SELECT
    tc.type_nom,
    COUNT(ch.id_chambre) AS nombre_chambres
FROM TypeChambre tc
LEFT JOIN Chambre ch ON tc.id_type = ch.id_type
GROUP BY tc.id_type, tc.type_nom;
```

Cette requête affiche le nombre de chambres disponibles pour chaque type (simple, double, etc.).

- 
- **Afficher la liste des chambres qui ne sont pas réservées pour une période donnée (exemple : du 2025-07-01 au 2025-07-31)**

```
SELECT
```

```
    ch.id_chambre,
```

```
    ch.numero_chambre,
```

```
    h.ville AS hotel_ville,
```

```
    tc.type_nom AS type_chambre
```

```
FROM Chambre ch
```

```
JOIN Hotel h ON ch.id_hotel = h.id_hotel
```

```
JOIN TypeChambre tc ON ch.id_type = tc.id_type
```

```
WHERE ch.id_chambre NOT IN (
```

```
    SELECT DISTINCT rc.id_chambre
```

```
    FROM ReservationChambre rc
```

```
    JOIN Reservation r ON rc.id_reservation = r.id_reservation
```

```
    WHERE (r.date_arrivee <= '2025-07-31' AND r.date_depart >= '2025-07-01')
```

```
);
```

Cette requête affiche les chambres qui ne sont pas occupées pendant la période donnée. Elle exclut les chambres associées à des réservations chevauchant cette période.

---