

J'ai travaillé en monôme

Prénom : zakaria  
Nom : chargaoui  
Apogée : 2333554

Partie 1: Création et interrogation de la base de données

1. Utiliser MySQL workbench pour créer la base de données qui correspond au MCD.

Voire le script SQL « database.sql » pour créer la base de données selon le MCD fourni

2. Remplir les tables en vous appuyant sur les données fournies, en annexe, pour les tables Hotel, Client, Prestation, Type Chambre, Chambre, Réservation et Évaluation.

Voire le script SQL « insertion\_donnes.sql » pour insérer les donnes dans la base

3. Ecrire les requêtes SQL ET en algèbre relationnelle qui permettent de:

Pour les requêtes SQL ils existent dans le fichier « requêtes.sql»

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

$\Pi_{Id\_Reservation, Nom\_complet, Ville, Date\_arrivee, Date\_depart} (\sigma_{Reservation.Id\_Client = Client.Id\_Client \wedge Reservation.Id\_Reservation = Reservation\_Chambre.Id\_Reservation \wedge Reservation\_Chambre.Id\_Chambre = Chambre.Id\_Chambre \wedge Chambre.Id\_Hotel = Hotel.Id\_Hotel} (Reservation \times Client \times Reservation\_Chambre \times Chambre \times Hotel))$

b. Afficher les clients qui habitent à Paris.

$\sigma_{Ville = 'Paris'} (Client)$

c. Calculer le nombre de réservations faites par chaque client.

$\Pi_{Id\_Client, Nom\_complet; COUNT(Id\_Reservation) \rightarrow Nombre\_reservations} (Client \bowtie_{(client.Id\_Client = Reservation.Id\_Client)} Reservation)$

d. Donner le nombre de chambres pour chaque type de chambre.

$\Pi_{Type; COUNT(Id\_Chambre) \rightarrow Nombre\_chambres} (Type\_Chambre \bowtie_{(type\_Chambre.Id\_Type = chambre.Id\_Type)} Chambre)$

e. Afficher la liste des chambres qui ne sont pas réservées pour une période donnée (entre deux dates saisies par l'utilisateur).

```

Π Chambre.* (
  Chambre - Π Id_Chambre (
    σ (Date_arrivee ≤ '2025-06-20' ∧ Date_depart ≥ '2025-06-10') (
      Reservation_Chambre ⋈ (Reservation_Chambre.Id_Reservation = Reservation .Id_Reservation)
    )
  )
)

```

4. Qu'est ce que SQLite, quelle différence avec MySQL?

Différence entre SQLite et MySQL :

SQLite est un système de gestion de base de données relationnelle léger, sans serveur et autonome. Contrairement à MySQL qui est un système client-serveur, SQLite est intégré directement dans l'application qui l'utilise. Les principales différences sont:

**Architecture:** MySQL suit une architecture client-serveur, tandis que SQLite est une bibliothèque intégrée.

**Configuration:** MySQL nécessite une installation et une configuration, SQLite ne nécessite aucune configuration.

**Performance:** MySQL est plus performant pour les applications multi-utilisateurs et les gros volumes de données.

**Fonctionnalités:** MySQL offre plus de fonctionnalités (utilisateurs, permissions, procédures stockées).

**Portabilité:** Les bases SQLite sont contenues dans un seul fichier, très facile à déplacer.

**Utilisation:** SQLite est idéal pour les applications mobiles ou embarquées, MySQL pour les applications web.