



Mamourou
Anthony
Matthieu

SAÉ 1.04

Création d'une base de données

Table des matières

Présentation.....	3
En quoi consiste ce projet ?	3
Cahier des charges	3
Modèle logique de données (MLD)	4
Décomposition en entités	4
Attributs des entités.....	5
Dictionnaire de données.....	6
Modèle conceptuel de données (MCD)	11
Rendu graphique	11
Exportation au format SQL.....	12
Écriture de scripts de manipulation.....	13
Création d'une BDD de travail.....	13
Écriture du jeu d'essai	14
Ordre de remplissage	14
Contenu du script	14
Écriture des scripts de lecture et d'écriture.....	15
Annexe.....	16
Script exporté du MLD sous <i>MySQL Workbench 8.0</i> (BDD.sql)	16
Script du jeu d'essai de la base de données (JeuEssai.sql)	23
Scripts de manipulation	25
InscrireClient.sql	25
InscrireHebergeur.sql	25
CreeAvion.sql	25
CreeSiege.sql	25
CreeVol.sql	26
ReserverVol.sql	26
ReserverHebergement.sql	26
ReserverSejour.sql	26
CreeAppartMaison.sql	27
CreeHotel.sql	27
CreeChambreHotel.sql	28
CreeAppartHotel.sql	28
RelieAppartAAppartHotel.sql	28
RelieStudioAAppartHotel.sql	29

VoirAppartMaisonDeHebergeur.sql.....	29
VoirHotelsDeHebergeur.sql	29
VoirChambresDeHotel.sql	29
VoirAppartHotelDeHebergeur.sql	29
VoirClient.sql.....	29
VoirHebergeur.sql	29

Présentation

En quoi consiste ce projet ?

Ce projet consiste en la création d'une base de données ayant pour but d'aider l'agence de voyages « Little Booking » à gérer ses informations, telles que celles de leurs hébergements, clients ou encore réservations.

Ainsi, il nous a fallu :

- Créer un MCD pour la base de données
- Faire un dictionnaire de données à partir de celui-ci
- L'adapter en MLD dans *MySQL Workbench*
- Exporter le MLD au format SQL pour permettre une utilisation dans un système réel
- Écrire des scripts d'exemple de peuplement de la base de données
- Écrire des scripts permettant de manipuler la base de données

Au cours de ce rapport, nous passerons en revue la réalisation de chacun de ces points et détaillerons les choix effectués au cours de ceux-ci.

Cahier des charges

Les hébergements à prendre en compte sont :

- Les appartements ou maisons
- Les appart-hôtels (pouvant comprendre apparts ou studios)
- Les hôtels

Les utilisateurs peuvent être :

- Hébergeurs
- Clients

Un client peut réserver :

- Un vol
- Un hébergement
- Un séjour (comprends les deux sous la forme d'une formule)

Beaucoup de détails ont été donnés sur les propriétés de chaque élément, ceux-ci seront détaillés par la suite.

Modèle logique de données (MLD)

Décomposition en entités

Il a fallu décomposer le cahier des charges en plusieurs entités, classées par catégories :

- Hébergements :
 - AppartementMaison
 - AppartHôtel
 - HôtelStudio
 - HôtelAppart
 - COMPRENDSTUDIO (relie AppartHôtel et HôtelStudio en $n \leftrightarrow n$)
 - COMPRENDSAPPART (relie AppartHôtel et HôtelAppart en $n \leftrightarrow n$)
 - Hôtel
 - Chambre (relié en $n \leftrightarrow 1$ à Hôtel)
- Hébergeurs :
 - Hébergeur
 - Hébergement (relie Hébergeur avec soit un AppartementMaison, soit un AppartHôtel, soit un Hôtel en $1 \leftrightarrow n$)
- Clients :
 - Client
- Vols :
 - Vol
 - Avion (Relié en $n \leftrightarrow 1$ à Vol)
 - Siège (Relié en $n \leftrightarrow 1$ à Avion)
- Réservations
 - RéservationHébergement (Relié à un client et à un hébergement)
 - RéservationVol (Relié à un client et à un vol)
 - RéservationSéjour (Relié à un client, un hébergement et un vol)

Attributs des entités

Les entités, écrites et décrites sous forme de MLD, sont comme il suit :

- Hébergements :
 - AppartementMaison(nb_appmaison, nom, pays, adresse, emplacements_hebergables, nb_presonnes_hebergables, nb_sdb, superficie, equipements, parking, parking_payant, fumeur, animaux, enfants, fetes, commentaire, noms_photos)
 - AppartHôtel(nb_apphotel, nom, localite, adresse, equipements, horaires_arrivee, horaires_depart, fumeur, animaux, enfants, fetes, ptitdej, langues, douche_ou_baignoire, serviettes, seche_cheveux, micro_ondes, plaques_cuisson, ustensiles_cuisine, bouilloire, frigo, tarif)
 - HôtelStudio(nb_studio, nb_lit_simple, nb_lit_double, nb_personne, superficie, nom_photos)
 - HôtelAppart(nb_appart, emplacements_hebergables, nb_pers_hebergables, nb_sdb, superficie, nb_chambres, nom_photos)
 - COMPRENDSSUDIO(#nb_apphotel, #nb_studio)
 - COMPRENDSPART(#nb_apphotel, #nb_appart)
 - Hôtel(nb_hotel, nom, localite, adresse, nb_etoiles, nom_contact, num_tel, num_tel2, chaine_hoteliere, equipements, horaires_arrivee, horaires_depart, fumeur, animaux, enfants, fetes, ptitdej, langues, navette_aeroport, resto_sur_place, formule_resto, lits_appoint, tarif_lit_appoint)
 - Chambre(nb_chambre, #nb_hotel, type_chambre, superficie, vue, tarif, equipements)
- Hébergeurs :
 - Hébergeur(nb_hebergeur, nom_prenom, adresse, num_tel, email, mdp)
 - Hébergement(nb_hebergement, type, #nb_hebergeur, nb_unique_hbgmt)
- Clients :
 - Client(nb_client, nom_prenom, adresse, num_tel, email, mdp, date_naissance, sexe)
- Vols :
 - Vol(identifiant_vol, #nb_avion, aero_depart, aero_arrive, horaires_depart, horaires_arrive, tarif_eco, tarif_eco_hublot, tarif_prem, tarif_prem_hublot, bagages_inclus, tarif_ajout_un_bagage, tarif_ajout_deux_bagage, tarif_ajout_trois_bagage)
 - Avion(nb_avion, type, nb_sieges)
 - Siège(nb_siege, #nb_avion, prem_classe, cote_hublot)

- Réservations :
 - RéservationHébergement(#nb_client, #nb_unique_hbgmt, date_depart, date_arrivee, nb_adultes, nb_enfants, ville, pays, remboursable)
 - RéservationVol(#nb_client, #identifiant_vol, date_depart, date_arrivee, nb_voyageurs, aller_retour, prem_classe, avec_escale)
 - RéservationSéjour(#nb_client, #nb_unique_hbgmt, #identifiant_vol, formule)

Dictionnaire de données

Le MLD a ensuite pu être adapté en un dictionnaire de données, décrivant ainsi chaque attribut et notant son type.

Nom	Description	Type	Taille
-----	-------------	------	--------

AppartementMaison			
nb_appmaison	Identifiant de l'appartement-maison	Numérique	
nom	Nom	Texte	50
pays	Pays	Texte	50
adresse	Adresse	Texte	100
emplacements_hebergeables	Emplacements hébergeables	Texte	200
nb_personnes_hebergeables	Capacité de l'appartement en terme de personnes hébergeables	Numérique	
nb_sdb	Nombre de salles de bains	Numérique	
superficie	Superficie	Numérique	
equipements	Equipements disponibles	Texte	300
parking	Présence ou non d'un parking	Booléen	
parking_payant	Si parking, payant ou non	Booléen	
fumeur	Autorisation ou non de fumer	Booléen	
animaux	Autorisation des animaux ou non	Booléen	
enfants	Autorisation des enfants ou non	Booléen	
fetes	Autorisation des fêtes ou non	Booléen	
commentaire	Commentaire(s)	Texte	50
noms_photos	Noms des photos	Texte	50

AppartHôtel			
nb_apphotel	Identifiant de l'appart-hôtel	Numérique	50
nom	Nom	Texte	50
localite	Localité (Région ou pays)	Texte	50
adresse	Adresse	Texte	100
equipements	Equipements disponibles	Texte	300
horaires_arrivee	Heures d'arrivée dans l'appart-hôtel	Date	
horaires_depart	Heures de départ dans l'appart-hôtel	Date	
fumeur	Autorisation de fumer ou non	Booléen	
animaux	Autorisation des animaux ou non	Booléen	
enfants	Autorisation des enfants ou non	Booléen	
fetes	Autorisation ou non des fêtes	Booléen	
ptitdej	Petit déjeuner fourni ou non	Booléen	

langues	Langues parlées par le personnel	Texte	200
douche_ou_baignoire	Présence ou non d'une douche ou baignoire	Booléen	
serviettes	Présence de serviettes	Booléen	
seche_cheveux	Présence d'un sèche-cheveux	Booléen	
Micro_ondes	Présence d'un micro-onde	Booléen	
plaques_cuisson	Présence de plaques de cuisson	Booléen	
ustensiles_cuisine	Présence d'ustensiles de cuisines	Booléen	
Bouilloire	Présence d'une bouilloire	Booléen	
frigo	Présence d'un frigo	Booléen	
tarif	Prix à la nuit	Numérique	

HôtelAppart			
nb_appart	Identifiant de l'appartement	Numérique	
emplacements_hebergeables	Emplacements hébergeables	Texte	200
nb_personnes_hebergeables	Nombre de personnes hébergeables	Numérique	
nb_sdb	Nombre de salles de bain	Numérique	
superficie	Superficie	Numérique	
nb_chambres	Nombre de chambres	Numérique	
noms_photos	Noms des photos	Texte	200

HôtelStudio			
nb_studio	Identifiant du studio	Numérique	
nb_lit_simple	Nombre de lits simples	Numérique	
nb_lit_double	Nombre de lits doubles	Numérique	
nb_personne	Nombre de personnes que ce studio peut accueillir	Numérique	
superficie	Superficie	Numérique	
noms_photos	Noms des photos	Texte	200

COMPRENDSAPPART			
nb_apphotel	Identifiant de l'appart-hôtel correspondant	Numérique	
nb_appart	Identifiant de l'appartement correspondant	Numérique	

COMPRENDSSTUDIO			
nb_apphotel	Identifiant de l'appart-hôtel correspondant	Numérique	
nb_studio	Identifiant du studio correspondant	Numérique	

Chambre			
nb_chambre	Identifiant de la chambre	Numérique	
nb_hotel	Identifiant de l'hôtel correspondant	Numérique	
type_chambre	Type de chambre	Texte	50
superficie	Superficie	Numérique	
vue	Vue à la fenêtre de la chambre	Texte	50
tarif	Prix à la nuit	Numérique	
equipements	Equipements disponibles	Texte	300

Hôtel			
nb_hotel	Identifiant de l'hôtel	Numérique	
nom	Nom	Texte	50
localite	Localité (Région ou pays)	Texte	50
adresse	Adresse	Texte	100
nb_etoiles	Nombre d'étoiles de 1 à 5	Numérique	
nom_contact	Nom de la personne chargée de contact	Texte	50
num_tel	Numéro de téléphone	Texte	50
num_tel2	Numéro de téléphone secondaire	Texte	50
chaîne_hoteliere	Chaîne hôtelière	Texte	50
equipements	Equipements disponibles	Texte	50
horaires_arrivee	Heures d'arrivées dans l'hôtel	Heure	
horaires_depart	Heures de départ dans l'hôtel	Heure	
fumeur	Autorisation de fumer	Booléen	
animaux	Autorisation aux animaux	Booléen	
enfants	Autorisation aux enfants	Booléen	
fetes	Autorisation aux fêtes	Booléen	
ptitdej	Disponibilité du petit-déjeuner	Booléen	
langues	Langues parlées par le personnel	Texte	200
navette_aeroport	Disponibilité d'une navette entre l'hôtel et un aéroport	Booléen	
resto_sur_place	Restaurant sur place ou non	Booléen	
formule_resto	Formule de restauration	Texte	50
lits_appoint	Présence ou non de lits d'appoints	Booléen	
tarif_lit_appoint	Prix d'un lit d'appoint si présent	Numérique	

Hébergeur			
nb_hebergeur	Identifiant de l'hébergeur	Numérique	
nom_prenom	Nom et prénom	Texte	100
adresse	Adresse	Texte	100
num_tel	Numéro de téléphone	Texte	50
email	Adresse e-mail	Texte	320
mdp	Mot de passe (hashé)	Texte	100

Hébergement			
nb_unique_hbgmt	Identifiant unique de l'hébergement	Numérique	
nb_hebergement	Numéro de l'hébergement correspondant	Numérique	
type	Type de l'hébergement correspondant	Texte	20
nb_hebergeur	Identifiant de l'hébergeur à qui appartient l'hébergement	Numérique	

Client			
nb_client	Identifiant du client	Numérique	
nom_prenom	Nom et prénom	Texte	50
adresse	Adresse	Texte	50
num_tel	Numéro de téléphone	Numérique	
email	Adresse e-mail	Texte	50
mdp	Mot de passe (hashé)	Texte	50

Date_naissance	Date de naissance	Date	
sexe	Genre	Texte	50

Vol			
identifiant_vol	Identifiant du vol	Texte	20
nb_avion	Identifiant de l'avion correspondant	Numérique	
aero_depart	Aéroport de départ	Texte	50
aero_arrive	Aéroport d'arrivée	Texte	50
horaires_depart	Heure de départ	Heure	
horaires_arrive	Heure d'arrivée	Heure	
tarif_eco	Prix en classe économique	Numérique	
tarif_eco_hublot	Prix en classe économique à côté du hublot	Numérique	
tarif_prem	Prix en première classe	Numérique	
tarif_prem_hublot	Prix en première place à côté du hublot	Numérique	
Bagages_inclus	Bagages inclus dans l'avion ou non	Booléen	
tarif_ajout_un_bagage	Prix pour ajouter un bagage	Numérique	
tarif_ajout_deux_bagages	Prix pour ajouter deux bagages	Numérique	
tarif_ajout_trois_bagages	Prix pour ajouter trois bagages	Numérique	

Avion			
nb_avion	Identifiant de l'avion	Numérique	
type	Type d'avion	Texte	50
nb_sieges	Nombre de sièges	Numérique	

Siège			
nb_siège	Identifiant du siège	Numérique	
nb_avion	Identifiant de l'avion correspondant	Numérique	
prem_classe	Siège de première classe on non	Booléen	
cote_hublo	Siège à côté du hublot ou non	Booléen	

RéservationHébergement			
nb_client	Identifiant du client correspondant	Numérique	
nb_unique_hbgmt	Identifiant de l'hébergement correspondant	Numérique	
date_arrivee	Date d'arrivée à l'hébergement	Date	
date_depart	Date du départ de l'hébergement	Date	
nb_adultes	Nombres d'adultes	Numérique	
nb_enfants	Nombre d'enfant	Numérique	
ville	Nom de la ville	Texte	50
pays	Nom du pays	Texte	50
remboursable	Réservation remboursable ou non	Booléen	

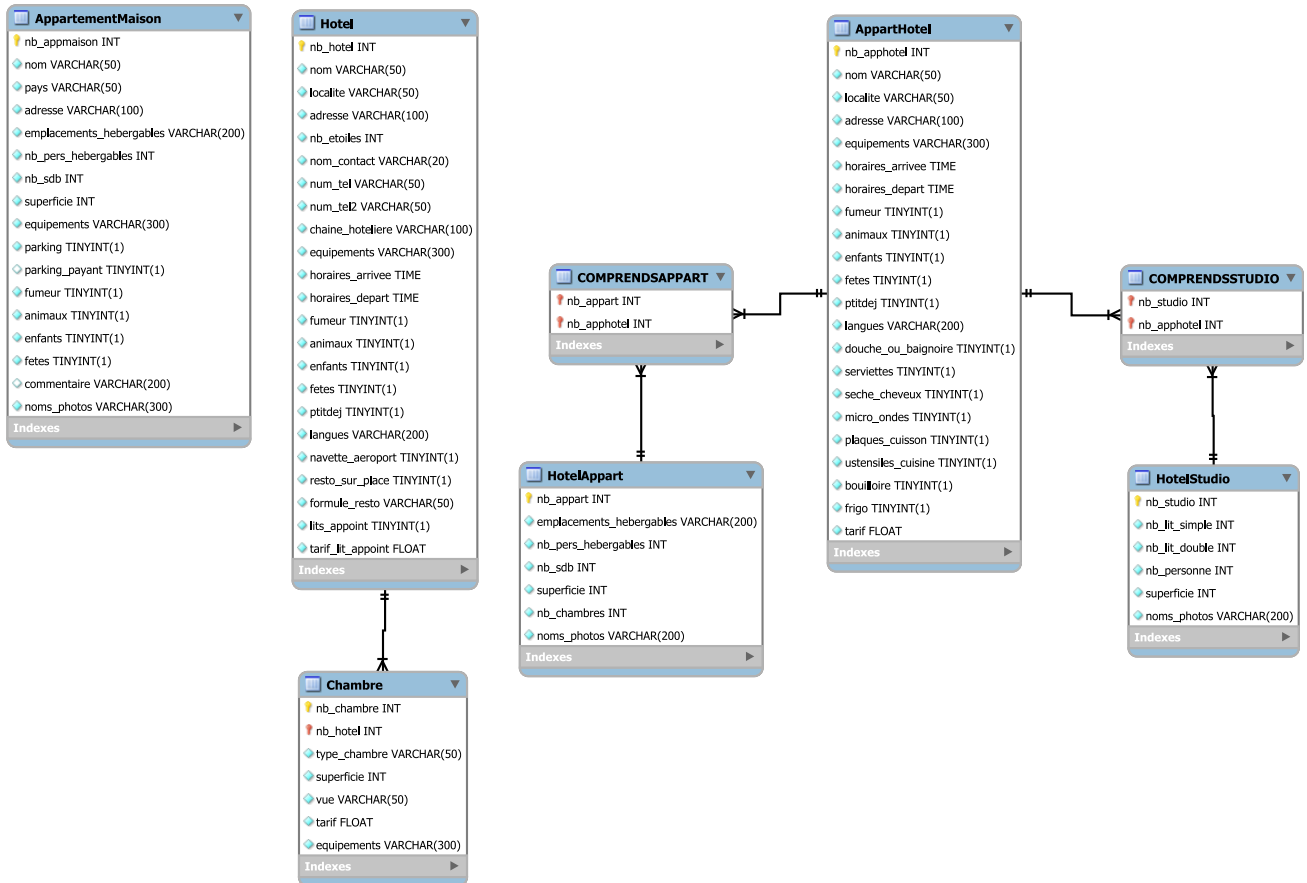
RéservationVol			
nb_client	Identifiant du client correspondant	Numérique	
identifiant_vol	Identifiant du vol correspondant	Numérique	
date_depart	Date de décollage	Date	
date_arrivee	Date d'arrivée de l'avion	Date	
nb_voyageurs	Nombre de voyageurs	Numérique	

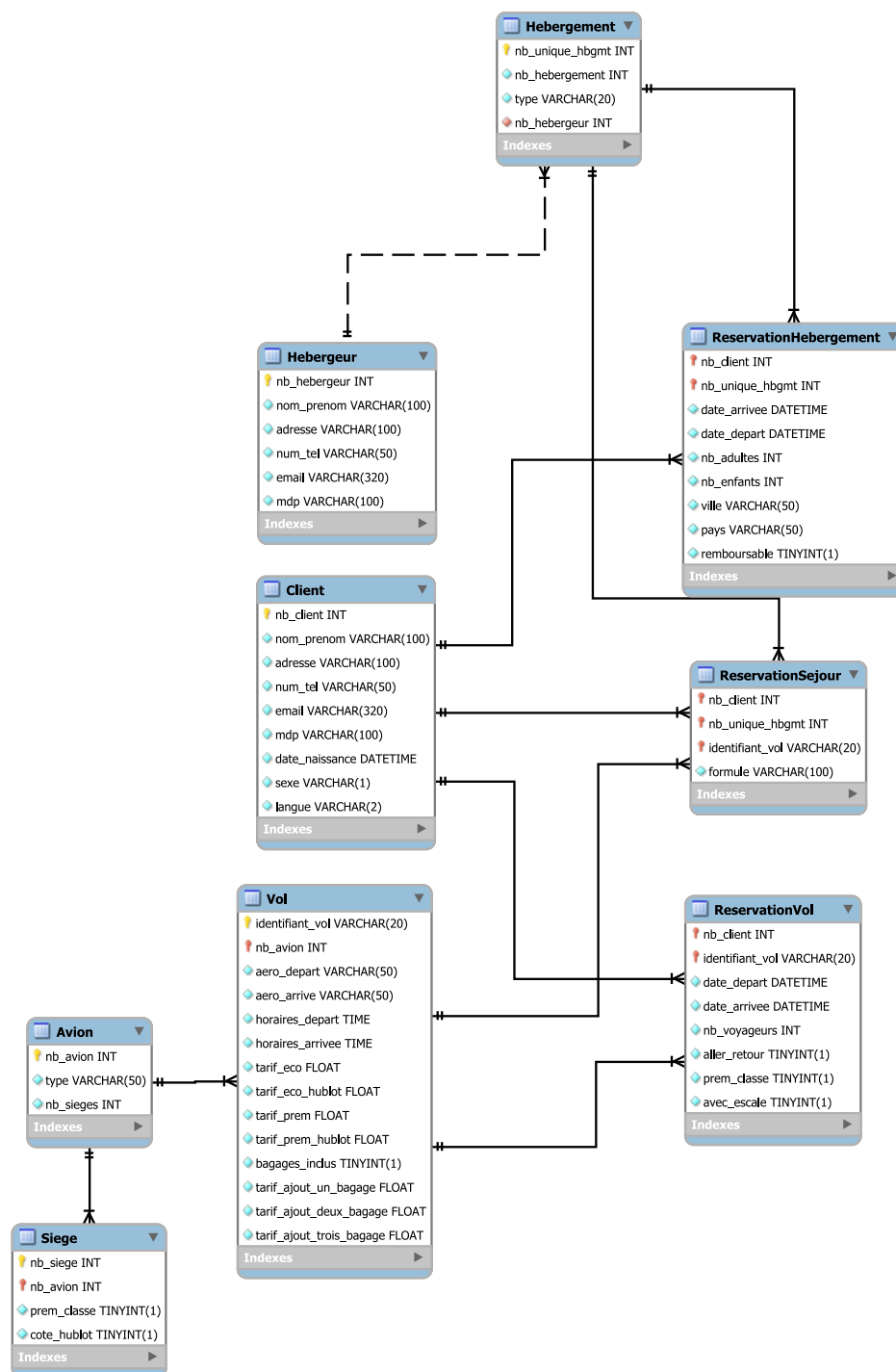
aller_retour	Allé-retour ou non	Booléen	
prem_classe	Assis en première classe ou non	Booléen	
avec_escale	Avec escale ou non	Booléen	
RéservationSéjour			
nb_client	Identifiant du client correspondant	Numérique	
nb_unique_hbgmt	Identifiant de l'hébergement correspondant	Numérique	
identifiant_vol	Identifiant du vol correspondant	Numérique	
formule	Formule	Texte	100

Modèle conceptuel de données (MCD)

Rendu graphique

Le MCD représente la version graphique des entités du MLD avec leurs propriétés du dictionnaire de données, et se présente comme il suit :





Exportation au format SQL

Le MCD ayant été créé avec *MySQL Workbench 8.0*, nous avons pu l'exporter en un script SQL permettant la recréation de ces tables et de leurs relations dans une base de données MySQL.

Le script exporté se trouve en annexe sous le nom de **BDD . sql**.

Attention : MySQL exporte la base sous forme d'un schéma (ici du nom de « littlebooking »). Il faut donc penser à prendre en compte ce nom dans les scripts d'insertion dans la base de données.

Écriture de scripts de manipulation

Création d'une BDD de travail

Il nous a fallu écrire des scripts afin de manipuler la base de données.

Mais avant cela, il nous faut un jeu d'essai : une collection d'entrées pour chacune de nos tables afin de pouvoir faire des requêtes de sélection.

Pour les écrire, commençons déjà par ouvrir notre terminal MySQL et à créer une base de données :

```
CREATE DATABASE voyages;  
USE voyages;
```

Maintenant, sourçons notre script :

```
SOURCE BDD.sql
```

Nous ne pouvons constater aucune erreur dans le terminal :

```
mysql> SOURCE BDD.sql  
Query OK, 0 rows affected (0.00 sec)  
  
Query OK, 0 rows affected (0.00 sec)  
  
Query OK, 0 rows affected, 1 warning (0.00 sec)  
  
Query OK, 1 row affected (0.06 sec)  
  
Database changed  
Query OK, 0 rows affected (0.52 sec)  
  
Query OK, 0 rows affected (0.43 sec)  
  
Query OK, 0 rows affected (0.49 sec)  
  
Query OK, 0 rows affected (0.48 sec)  
  
Query OK, 0 rows affected (0.50 sec)  
  
Query OK, 0 rows affected (0.54 sec)  
  
Query OK, 0 rows affected (0.51 sec)  
  
Query OK, 0 rows affected (0.52 sec)  
  
Query OK, 0 rows affected (0.47 sec)  
  
Query OK, 0 rows affected (0.52 sec)  
  
Query OK, 0 rows affected (0.50 sec)  
  
Query OK, 0 rows affected (0.50 sec)  
  
Query OK, 0 rows affected (0.48 sec)  
  
Query OK, 0 rows affected (0.47 sec)  
  
Query OK, 0 rows affected (0.76 sec)  
  
Query OK, 0 rows affected (0.81 sec)  
  
Query OK, 0 rows affected (0.51 sec)  
  
Query OK, 0 rows affected, 2 warnings (0.00 sec)  
  
Query OK, 0 rows affected (0.00 sec)  
  
Query OK, 0 rows affected (0.00 sec)  
  
mysql>
```

Les avertissements (« warnings ») sont dus à une dépréciation du système de changements de modes SQL, demandés par les requêtes **SET** du script généré par *MySQL Workbench 8.0*.

Une mise à jour de celui-ci peut probablement régler ce problème.

Maintenant que notre base de données est prête, nous sommes prêts à y écrire des requêtes d'insertions à transvaser en script.

Écriture du jeu d'essai

Ordre de remplissage

L'ordre est un élément capital de l'écriture du jeu d'essai : si l'on essaie d'insérer des données dans une table qui comporte des clés primaires référençant des tables qui n'ont aucune données, une erreur nous sera remontée.

Il est donc important de lister l'ordre dans lequel les différentes tables doivent être remplies, afin d'éviter ce problème.

Il faut donc remplir les tables dans l'ordre suivant :

Ordre	Tables	Description
1	AppartementMaison Hôtel AppartHôtel HôtelAppart HôtelStudio	Les hébergements
2	Chambre COMPRENDSSTUDIO COMPRENDSAPPART	Les sous-tables des hébergements
3	Avion	Les avions
4	Siège Vol	Les vols et les sièges des avions
5	Hébergeur Client	Les hébergeurs et clients
6	Hébergement	Les pairs pour les hébergements
7	RéservationHébergement RéservationVol RéservationSéjour	Les réservations

Contenu du script

Le script du jeu d'essai est disponible en annexe sous le nom de **JeuEssai.sql**.

Celui-ci comprend :

- 3 Appartements-Maison
- 2 Hôtels (de respectivement 2 et 3 chambres)
- 2 AppartHôtel (l'un avec un appart et l'autre avec un studio)
- 4 Avions (de respectivement 1, 2, 3 et 4 sièges)
- 2 Vols (chacun lié à un avion)
- 6 hébergeurs (l'un possède six hébergements, un autre en possède un, les autres n'ont pas encore enregistré les leurs)
- 10 clients
- 3 réservations (une pour chaque type de réservation)

Écriture des scripts de lecture et d'écriture

Comme requis, nous avons écrit des scripts permettant la manipulation de la base de données.

Vous trouverez en annexe les scripts suivants :

- **InscrireClient.sql** : inscrit un Client
- **InscrireHebergeur.sql** : inscrit un Hébergeur

- **CreeAvion.sql** : créé un Avion,
- **CreeSiege.sql** : créé un Siège d'avion
- **CreeVol.sql** : créé un Vol d'avion

- **ReserverVol.sql** : réserve un Vol pour un Client
- **ReserverHebergement.sql** : réserve un Hébergement pour un Client
- **ReserverSejour.sql** : réserve un séjour pour un Client

- **CreeAppartMaison.sql** : créé un AppartementMaison pour un Hébergeur (en créant un Hébergement reliant les deux)
- **CreeHotel.sql** : créé un Hôtel pour un Hébergeur (en créant un Hébergement reliant les deux)
- **CreeChambreHotel.sql** : créée une chambre et la relie à un Hôtel
- **CreeAppartHotel.sql** : créé un AppartHôtel pour un Hébergeur (en créant un Hébergement reliant les deux)
- **RelieAppartAAppartHotel.sql** : créé un HôtelAppart et le relie à un AppartHôtel (en utilisant un COMPRENDSPART)
- **RelieStudioAAppartHotel.sql** : créé un HôtelStudio et le relie à un AppartHôtel (en utilisant un COMPRENDSTUDIO)

- **VoirAppartMaisonDeHebergeur.sql** : affiche les AppartementMaison possédés par un hébergeur
- **VoirHotelsDeHebergeur.sql** : affiche les Hôtel possédés par un hébergeur
- **VoirChambresDeHotel.sql** : affiche les Chambre d'un hôtel
- **VoirAppartHotelDeHebergeur.sql** : affiche les AppartHôtel possédés par un hébergeur
- **VoirClient.sql** : affiche les informations d'un Client
- **VoirHebergeur.sql** : affiche les informations d'un Hébergeur

Annexe

Script exporté du MLD sous *MySQL Workbench 8.0* (BDD .sql)

```
-- MySQL Script generated by MySQL Workbench
-- Sat Jan 21 20:45:21 2023
-- Model: Sakila Full      Version: 2.0
-- MySQL Workbench Forward Engineering

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='TRADITIONAL,ALLOW_INVALID_DATES';

-- 
-- Schema littlebooking
-- 
-- 
-- Schema littlebooking
-- 

CREATE SCHEMA IF NOT EXISTS `littlebooking` ;
USE `littlebooking` ;

-- 
-- Table `littlebooking`.`AppartementMaison`
-- 
CREATE TABLE IF NOT EXISTS `littlebooking`.`AppartementMaison` (
  `nb_appmaison` INT NOT NULL,
  `nom` VARCHAR(50) NOT NULL,
  `pays` VARCHAR(50) NOT NULL,
  `adresse` VARCHAR(100) NOT NULL,
  `emplacements_hebergables` VARCHAR(200) NOT NULL,
  `nb_pers_hebergables` INT NOT NULL DEFAULT 1,
  `nb_sdb` INT NOT NULL DEFAULT 1,
  `superficie` INT NOT NULL COMMENT 'En m²',
  `equipements` VARCHAR(300) NOT NULL COMMENT 'Disponibles:\n- Cuisine et entretien: Cuisine, kitchenette, lave-linge\n- Divertissement: Television, piscine, mini-bar sauna\n- Vue et exterieur: balcon, vue sur jardin, terrasse\n- Autres commodites: climatiseur, chauffage ou connexion wifi',
  `parking` TINYINT(1) NOT NULL DEFAULT 0,
  `parking_payant` TINYINT(1) NULL DEFAULT 0,
  `fumeur` TINYINT(1) NOT NULL,
  `animaux` TINYINT(1) NOT NULL,
  `enfants` TINYINT(1) NOT NULL,
  `fetes` TINYINT(1) NOT NULL,
  `commentaire` VARCHAR(200) NULL DEFAULT NULL,
  `noms_photos` VARCHAR(300) NOT NULL COMMENT 'Separer les noms des fichiers par "|"\"nMinimum 1, maximum 5',
  PRIMARY KEY (`nb_appmaison`),
  UNIQUE INDEX `nb_appmaison_UNIQUE` (`nb_appmaison` ASC))
ENGINE = InnoDB;

-- 
-- Table `littlebooking`.`AppartHotel`
-- 
CREATE TABLE IF NOT EXISTS `littlebooking`.`AppartHotel` (
  `nb_apphotel` INT NOT NULL,
  `nom` VARCHAR(50) NOT NULL,
  `localite` VARCHAR(50) NOT NULL,
  `adresse` VARCHAR(100) NOT NULL,
  `equipements` VARCHAR(300) NOT NULL COMMENT 'Disponible: Wi-Fi gratuit, Parking gratuit, Climatisation, Terrasse, Jardin, Chauffage, Reception 24h/24, Bar, Piscine, Sauna, Jacuzzi',
  `horaires_arrivee` TIME NOT NULL,
  `horaires_depart` TIME NOT NULL,
  `fumeur` TINYINT(1) NOT NULL,
  `animaux` TINYINT(1) NOT NULL,
```

```

`enfants` TINYINT(1) NOT NULL,
`fetes` TINYINT(1) NOT NULL,
`ptitdej` TINYINT(1) NOT NULL,
`langues` VARCHAR(200) NOT NULL,
`douche_ou_baignoire` TINYINT(1) NOT NULL,
`serviettes` TINYINT(1) NOT NULL,
`seche_cheveux` TINYINT(1) NOT NULL,
`micro_ondes` TINYINT(1) NOT NULL,
`plaques_cuisson` TINYINT(1) NOT NULL,
`ustensiles_cuisine` TINYINT(1) NOT NULL,
`bouilloire` TINYINT(1) NOT NULL,
`frigo` TINYINT(1) NOT NULL,
`tarif` FLOAT NOT NULL COMMENT 'En Euros (€)',
PRIMARY KEY (`nb_apphotel`),
UNIQUE INDEX `nb_apphotel_UNIQUE` (`nb_apphotel` ASC))
ENGINE = InnoDB;

-----
-- Table `littlebooking`.`HotelStudio`
-----
CREATE TABLE IF NOT EXISTS `littlebooking`.`HotelStudio` (
  `nb_studio` INT NOT NULL AUTO_INCREMENT,
  `nb_lit_simple` INT NOT NULL,
  `nb_lit_double` INT NOT NULL,
  `nb_personne` INT NOT NULL,
  `superficie` INT NOT NULL COMMENT 'En m²',
  `noms_photos` VARCHAR(200) NOT NULL COMMENT 'Separer les noms des fichiers par "|"\"Maximum
3',
  PRIMARY KEY (`nb_studio`),
  UNIQUE INDEX `nb_studio_UNIQUE` (`nb_studio` ASC))
ENGINE = InnoDB;

-----
-- Table `littlebooking`.`HotelAppart`
-----
CREATE TABLE IF NOT EXISTS `littlebooking`.`HotelAppart` (
  `nb_appart` INT NOT NULL AUTO_INCREMENT,
  `emplacements_hebergables` VARCHAR(200) NOT NULL,
  `nb_pers_hebergables` INT NOT NULL DEFAULT 1,
  `nb_sdb` INT NOT NULL DEFAULT 1,
  `superficie` INT NOT NULL COMMENT 'En m²',
  `nb_chambres` INT NOT NULL,
  `noms_photos` VARCHAR(200) NOT NULL COMMENT 'Separer les noms des fichiers par "|"\"Maximum
3',
  PRIMARY KEY (`nb_appart`),
  UNIQUE INDEX `nb_appart_UNIQUE` (`nb_appart` ASC))
ENGINE = InnoDB;

-----
-- Table `littlebooking`.`COMPRENDSAPPART`
-----
CREATE TABLE IF NOT EXISTS `littlebooking`.`COMPRENDSAPPART` (
  `nb_appart` INT NOT NULL,
  `nb_apphotel` INT NOT NULL,
  PRIMARY KEY (`nb_appart`, `nb_apphotel`),
  INDEX `fk_COMPRENDSAPPART_AppartHotel1_idx` (`nb_apphotel` ASC),
  INDEX `fk_COMPRENDSAPPART_HotelAppart_idx` (`nb_appart` ASC),
  CONSTRAINT `fk_COMPRENDSAPPART_AppartHotel1`
    FOREIGN KEY (`nb_apphotel`)
    REFERENCES `littlebooking`.`AppartHotel` (`nb_apphotel`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_COMPRENDSAPPART_HotelAppart`
    FOREIGN KEY (`nb_appart`)

```

```

REFERENCES `littlebooking`.`HotelAppart` (`nb_appart`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;

-----
-- Table `littlebooking`.`COMPRENDSSTUDIO`
-----
CREATE TABLE IF NOT EXISTS `littlebooking`.`COMPRENDSSTUDIO` (
  `nb_studio` INT NOT NULL,
  `nb_apphotel` INT NOT NULL,
  PRIMARY KEY (`nb_studio`, `nb_apphotel`),
  INDEX `fk_COMPRENDSSTUDIO_AppartHotel1_idx` (`nb_apphotel` ASC),
  INDEX `fk_COMPRENDSSTUDIO_HotelStudio1_idx` (`nb_studio` ASC),
  CONSTRAINT `fk_COMPRENDSSTUDIO_AppartHotel1`
    FOREIGN KEY (`nb_apphotel`)
    REFERENCES `littlebooking`.`AppartHotel` (`nb_apphotel`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_COMPRENDSSTUDIO_HotelStudio1`
    FOREIGN KEY (`nb_studio`)
    REFERENCES `littlebooking`.`HotelStudio` (`nb_studio`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

-----
-- Table `littlebooking`.`Hotel`
-----
CREATE TABLE IF NOT EXISTS `littlebooking`.`Hotel` (
  `nb_hotel` INT NOT NULL AUTO_INCREMENT,
  `nom` VARCHAR(50) NOT NULL,
  `localite` VARCHAR(50) NOT NULL,
  `adresse` VARCHAR(100) NOT NULL,
  `nb_etoiles` INT NOT NULL,
  `nom_contact` VARCHAR(20) NOT NULL,
  `num_tel` VARCHAR(50) NOT NULL COMMENT 'Au format international',
  `num_tel2` VARCHAR(50) NOT NULL COMMENT 'Au format international',
  `chaine_hoteliere` VARCHAR(100) NOT NULL,
  `equipements` VARCHAR(300) NOT NULL COMMENT 'Disponible: Wi-Fi gratuit, Parking gratuit,
Climatisation, Terrasse, Jardin, Chauffage, Reception 24h/24, Bar, Piscine, Sauna, Jacuzzi',
  `horaires_arrivee` TIME NOT NULL,
  `horaires_depart` TIME NOT NULL,
  `fumeur` TINYINT(1) NOT NULL,
  `animaux` TINYINT(1) NOT NULL,
  `enfants` TINYINT(1) NOT NULL,
  `fetes` TINYINT(1) NOT NULL,
  `ptitdej` TINYINT(1) NOT NULL,
  `langues` VARCHAR(200) NOT NULL,
  `navette_aeroport` TINYINT(1) NOT NULL,
  `resto_sur_place` TINYINT(1) NOT NULL,
  `formule_resto` VARCHAR(50) NOT NULL,
  `lits_appoint` TINYINT(1) NOT NULL,
  `tarif_lit_appoint` FLOAT NOT NULL COMMENT 'En Euros (€)',
  PRIMARY KEY (`nb_hotel`),
  UNIQUE INDEX `nom_hotel_UNIQUE` (`nom` ASC))
ENGINE = InnoDB;

-----
-- Table `littlebooking`.`Chambre`
-----
CREATE TABLE IF NOT EXISTS `littlebooking`.`Chambre` (
  `nb_chambre` INT NOT NULL AUTO_INCREMENT,
  `nb_hotel` INT NOT NULL,

```

```

`type_chambre` VARCHAR(50) NOT NULL,
`superficie` INT NOT NULL COMMENT 'En m²',
`vue` VARCHAR(50) NOT NULL,
`tarif` FLOAT NOT NULL COMMENT 'En Euros (€)',
`equipements` VARCHAR(300) NOT NULL COMMENT 'Disponible: Baignoire, Television, bouilloire
electrique, balcon, vue, terrasse',
PRIMARY KEY (`nb_chambre`, `nb_hotel`),
UNIQUE INDEX `nb_chambre_UNIQUE` (`nb_chambre` ASC),
INDEX `fk_Chambre_Hotell_idx` (`nb_chambre` ASC),
INDEX `fk_Chambre_Hotell` (`nb_hotel` ASC),
CONSTRAINT `fk_Chambre_Hotell`
  FOREIGN KEY (`nb_hotel`)
  REFERENCES `littlebooking`.`Hotel` (`nb_hotel`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB;

-----
-- Table `littlebooking`.`Hebergeur`
-----
CREATE TABLE IF NOT EXISTS `littlebooking`.`Hebergeur` (
  `nb_hebergeur` INT NOT NULL AUTO_INCREMENT,
  `nom_prenom` VARCHAR(100) NOT NULL,
  `adresse` VARCHAR(100) NOT NULL,
  `num_tel` VARCHAR(50) NOT NULL COMMENT 'Au format international',
  `email` VARCHAR(320) NOT NULL,
  `mdp` VARCHAR(100) NOT NULL COMMENT 'Sous la forme d\'un hash SHA-256',
  PRIMARY KEY (`nb_hebergeur`),
  UNIQUE INDEX `nb_hebergeur_UNIQUE` (`nb_hebergeur` ASC))
ENGINE = InnoDB;

-----
-- Table `littlebooking`.`Client`
-----
CREATE TABLE IF NOT EXISTS `littlebooking`.`Client` (
  `nb_client` INT NOT NULL AUTO_INCREMENT,
  `nom_prenom` VARCHAR(100) NOT NULL,
  `adresse` VARCHAR(100) NOT NULL,
  `num_tel` VARCHAR(50) NOT NULL COMMENT 'Au format international',
  `email` VARCHAR(320) NOT NULL,
  `mdp` VARCHAR(100) NOT NULL COMMENT 'Sous la forme d\'un hash SHA-256',
  `date_naissance` DATETIME NOT NULL,
  `sexe` VARCHAR(1) NOT NULL COMMENT 'M: Masculin\nF: Feminin\nA: Autre',
  `langue` VARCHAR(2) NOT NULL COMMENT 'Sous la forme d\'un code ISO 639-1',
  PRIMARY KEY (`nb_client`),
  UNIQUE INDEX `nb_client_UNIQUE` (`nb_client` ASC))
ENGINE = InnoDB;

-----
-- Table `littlebooking`.`Avion`
-----
CREATE TABLE IF NOT EXISTS `littlebooking`.`Avion` (
  `nb_avion` INT NOT NULL AUTO_INCREMENT,
  `type` VARCHAR(50) NOT NULL,
  `nb_sieges` INT NOT NULL,
  PRIMARY KEY (`nb_avion`),
  UNIQUE INDEX `nb_avion_UNIQUE` (`nb_avion` ASC))
ENGINE = InnoDB;

-----
-- Table `littlebooking`.`Vol`
-----
CREATE TABLE IF NOT EXISTS `littlebooking`.`Vol` (

```

```

`identifiant_vol` VARCHAR(20) NOT NULL COMMENT 'Sous la forme \"XXXX XXXX XXXX XXXX\"',
`nb_avion` INT NOT NULL,
`aero_depart` VARCHAR(50) NOT NULL,
`aero_arrivee` VARCHAR(50) NOT NULL,
`horaires_depart` TIME NOT NULL,
`horaires_arrivee` TIME NOT NULL,
`tarif_eco` FLOAT NOT NULL COMMENT 'En Euros (€)',
`tarif_eco_hublot` FLOAT NOT NULL COMMENT 'En Euros (€)',
`tarif_prem` FLOAT NOT NULL COMMENT 'En Euros (€)',
`tarif_prem_hublot` FLOAT NOT NULL COMMENT 'En Euros (€)',
`bagages_inclus` TINYINT(1) NOT NULL,
`tarif_ajout_un_bagage` FLOAT NOT NULL COMMENT 'En Euros (€)',
`tarif_ajout_deux_bagage` FLOAT NOT NULL COMMENT 'En Euros (€)',
`tarif_ajout_trois_bagage` FLOAT NOT NULL COMMENT 'En Euros (€)',
PRIMARY KEY (`identifiant_vol`, `nb_avion`),
UNIQUE INDEX `identifiant_vol_UNIQUE` (`identifiant_vol` ASC),
INDEX `fk_Vol_Avion1_idx` (`nb_avion` ASC),
CONSTRAINT `fk_Vol_Avion1`
  FOREIGN KEY (`nb_avion`)
    REFERENCES `littlebooking`.`Avion` (`nb_avion`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB;

-----
-- Table `littlebooking`.`Siege`
-----

CREATE TABLE IF NOT EXISTS `littlebooking`.`Siege` (
  `nb_siege` INT NOT NULL AUTO_INCREMENT,
  `nb_avion` INT NOT NULL,
  `prem_classe` TINYINT(1) NOT NULL,
  `cote_hublot` TINYINT(1) NOT NULL,
  PRIMARY KEY (`nb_siege`, `nb_avion`),
  INDEX `fk_Siege_Avion1_idx` (`nb_avion` ASC),
  CONSTRAINT `fk_Siege_Avion1`
    FOREIGN KEY (`nb_avion`)
      REFERENCES `littlebooking`.`Avion` (`nb_avion`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

-----
-- Table `littlebooking`.`Hebergement`
-----

CREATE TABLE IF NOT EXISTS `littlebooking`.`Hebergement` (
  `nb_unique_hbgmt` INT UNSIGNED NOT NULL AUTO_INCREMENT,
  `nb_hebergement` INT NOT NULL COMMENT 'Selon le type, soit nb_hotel, soit nb_appmaison, soit nb_apphotel.',
  `type` VARCHAR(20) NOT NULL COMMENT 'Possibilites: HOTEL, APPMAISON, APPHOTEL',
  `nb_hebergeur` INT NOT NULL,
  PRIMARY KEY (`nb_unique_hbgmt`),
  INDEX `fk_Hebergement_Hebergeur1_idx` (`nb_hebergeur` ASC),
  CONSTRAINT `fk_Hebergement_Hebergeur1`
    FOREIGN KEY (`nb_hebergeur`)
      REFERENCES `littlebooking`.`Hebergeur` (`nb_hebergeur`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

-----
-- Table `littlebooking`.`ReservationHebergement`
-----

CREATE TABLE IF NOT EXISTS `littlebooking`.`ReservationHebergement` (
  `nb_client` INT NOT NULL,

```

```

`nb_unique_hbgmt` INT UNSIGNED NOT NULL,
`date_arrivee` DATETIME NOT NULL,
`date_depart` DATETIME NOT NULL,
`nb_adultes` INT NOT NULL,
`nb_enfants` INT NOT NULL,
`ville` VARCHAR(50) NOT NULL,
`pays` VARCHAR(50) NOT NULL,
`remboursable` TINYINT(1) NOT NULL,
INDEX `fk_ReservationHebergement_Client1_idx` (`nb_client` ASC),
PRIMARY KEY (`nb_client`, `nb_unique_hbgmt`),
INDEX `fk_ReservationHebergement_Hebergeur1_idx` (`nb_unique_hbgmt` ASC),
CONSTRAINT `fk_ReservationHebergement_Client1`
  FOREIGN KEY (`nb_client`)
  REFERENCES `littlebooking`.`Client` (`nb_client`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION,
CONSTRAINT `fk_ReservationHebergement_Hebergeur1`
  FOREIGN KEY (`nb_unique_hbgmt`)
  REFERENCES `littlebooking`.`Hebergement` (`nb_unique_hbgmt`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-----
-- Table `littlebooking`.`ReservationVol`
-----

```

```

CREATE TABLE IF NOT EXISTS `littlebooking`.`ReservationVol` (
  `nb_client` INT NOT NULL,
  `identifiant_vol` VARCHAR(20) NOT NULL,
  `date_depart` DATETIME NOT NULL,
  `date_arrivee` DATETIME NOT NULL,
  `nb_voyageurs` INT NOT NULL,
  `aller_retour` TINYINT(1) NOT NULL,
  `prem_classe` TINYINT(1) NOT NULL,
  `avec_escale` TINYINT(1) NOT NULL,
  PRIMARY KEY (`nb_client`, `identifiant_vol`),
  INDEX `fk_ReservationVol_Client1_idx` (`nb_client` ASC),
  INDEX `fk_ReservationVol_Vol1_idx` (`identifiant_vol` ASC),
  CONSTRAINT `fk_ReservationVol_Client1`
    FOREIGN KEY (`nb_client`)
    REFERENCES `littlebooking`.`Client` (`nb_client`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_ReservationVol_Vol1`
    FOREIGN KEY (`identifiant_vol`)
    REFERENCES `littlebooking`.`Vol` (`identifiant_vol`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-----
-- Table `littlebooking`.`ReservationSejour`
-----

```

```

CREATE TABLE IF NOT EXISTS `littlebooking`.`ReservationSejour` (
  `nb_client` INT NOT NULL,
  `nb_unique_hbgmt` INT UNSIGNED NOT NULL,
  `identifiant_vol` VARCHAR(20) NOT NULL,
  `formule` VARCHAR(100) NOT NULL,
  PRIMARY KEY (`nb_client`, `nb_unique_hbgmt`, `identifiant_vol`),
  INDEX `fk_ReservationSejour_Client1_idx` (`nb_client` ASC),
  INDEX `fk_ReservationSejour_Vol1_idx` (`identifiant_vol` ASC),
  INDEX `fk_ReservationSejour_Hebergeur1_idx` (`nb_unique_hbgmt` ASC),
  CONSTRAINT `fk_ReservationSejour_Client1`
    FOREIGN KEY (`nb_client`)
    REFERENCES `littlebooking`.`Client` (`nb_client`)

```

```

    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
CONSTRAINT `fk_ReservationSejour_Hebergeur1`
    FOREIGN KEY (`nb_unique_hbgmt`)
    REFERENCES `littlebooking`.`Hebergement` (`nb_unique_hbgmt`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
CONSTRAINT `fk_ReservationSejour_Vol1`
    FOREIGN KEY (`identifiant_vol`)
    REFERENCES `littlebooking`.`Vol` (`identifiant_vol`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;

```

Script du jeu d'essai de la base de données (JeuEssai.sql)

```
INSERT INTO AppartementMaison VALUES (1, "Maison de la chouette", "France", "3 rue des patates",
"Deux chambres, salon", 5, 1, 40, "Cuisine, Television, Wi-Fi", 1, 0, 0, 1, 1, 0, "Petite maison
a double etage dans un coin de foret", "maisonchouette-dehors.png|maisonchouette-dedans.png");
INSERT INTO AppartementMaison VALUES (2, "Logement coup-de-coeur", "France", "637 impasse du
vieux chene", "Une chambre", 1, 1, 20, "Television, Wi-Fi", 1, 1, 0, 0, 1, 1, "Logement
singulier presentant un bon rapport qualite/prix, pour tout type de personne",
"logementcdc.jpg");
INSERT INTO AppartementMaison VALUES (3, "Appartement etudiant", "France", "32 avenue du boeuf
de kobe", "Une chambre", 2, 0, 10, "Lave-linge, Wi-Fi", 0, 0, 1, 0, 0, 1, "Petit appartement
pour etudiant", "appartetu1.png|appartetu2.png|appartetu3.png");

INSERT INTO Hotel VALUES (1, "Hotel du beau soleil", "Bourgogne-Franche-Comte", "96 place du
sourire", 3, "Marco DIAZ", "+33612345678", "", "Groupe MIAM", "Wi-Fi gratuit, Reception 24h/24",
"04:00", "23:00", 0, 0, 1, 0, 1, "Francais, Anglais", 0, 1, "pension-complete", 0, 0.0);
INSERT INTO Hotel VALUES (2, "Hotel au petit ramonneur", "Normandie", "12 port du gros brochet",
1, "Bianca PITAINE", "+33386010203", "+33604050607", "Independant", "Chauffage, Bar", "06:00",
"20:00", 1, 1, 0, 0, 0, "Francais", 0, 1, "demi-pension", 1, 15.0);

INSERT INTO AppartHotel VALUES (1, "Logement du pic glace", "Alpes", "Chemin de terre du
moineau", "Wi-Fi gratuit, Chauffage, Sauna", "02:00", "22:00", 0, 1, 1, 0, 1, "Francais,
Anglais, Espagnol", 1, 1, 0, 1, 0, 1, 0, 1, 112.0);
INSERT INTO AppartHotel VALUES (2, "Logement de ville", "Allie", "1971 rue des fleurs",
"Terrasse, Jardin", "08:00", "20:00", 0, 0, 1, 0, 0, "Francais", 1, 0, 0, 1, 1, 0, 1, 1, 55.99);

INSERT INTO HotelAppart VALUES (1, "Deux chambres, salon", 5, 1, 20, 2, "picglaceapphotel.png");
INSERT INTO HotelStudio VALUES (1, 2, 0, 2, 20, "logvileapphotel.png");

INSERT INTO COMPRENDSAPPART VALUES (1, 1);
INSERT INTO COMPRENDSSTUDIO VALUES (1, 2);

INSERT INTO Chambre VALUES (1, 1, "Chambre classique", 20, "Vue sur la mer", 35.99, "Vue");
INSERT INTO Chambre VALUES (2, 1, "Chambre double", 40, "Vue sur la mer", 55.99, "Vue");
INSERT INTO Chambre VALUES (3, 2, "Chambre normale", 30, "Aucune", 70.0, "Television, Bouilloire
electrique");
INSERT INTO Chambre VALUES (4, 2, "Chambre superieure", 50, "Vue sur la ville", 90.0,
"Television, Bouilloire, Baignoire");
INSERT INTO Chambre VALUES (5, 2, "Chambre de luxe", 80, "Vue sur la ville", 110.0, "Television,
Bouilloire, Baignoire, Balcon");

INSERT INTO Avion VALUES (1, "A220-100", 1);
INSERT INTO Avion VALUES (2, "A319neo", 2);
INSERT INTO Avion VALUES (3, "A350 XWB-900", 3);
INSERT INTO Avion VALUES (4, "A330-200F", 4);

INSERT INTO Siege VALUES (1, 1, 1, 1);
INSERT INTO Siege VALUES (2, 2, 0, 0);
INSERT INTO Siege VALUES (3, 2, 1, 0);
INSERT INTO Siege VALUES (4, 3, 1, 1);
INSERT INTO Siege VALUES (5, 3, 0, 0);
INSERT INTO Siege VALUES (6, 3, 0, 1);
INSERT INTO Siege VALUES (7, 4, 0, 0);
INSERT INTO Siege VALUES (8, 4, 0, 1);
INSERT INTO Siege VALUES (9, 4, 1, 0);
INSERT INTO Siege VALUES (10, 4, 1, 1);

INSERT INTO Vol VALUES ("FRUS 0000 CPP4 0801", 1, "Paris", "New York", "10:00", "16:00", 30.0,
40.0, 50.0, 70.0, 0, 60.0, 110.0, 160.0);
INSERT INTO Vol VALUES ("FRJP 0000 MNGA 2109", 4, "Paris", "Tokyo", "6:00", "18:00", 60.0, 65.0,
115.0, 130.0, 1, 30.0, 55.0, 80.0);

INSERT INTO Hebergeur VALUES (1, "Monique DUPONT", "28 rue du beau charmeur", "+33908070605",
"moniquedupong@hotmail.com",
"856e407ff24fffc53b85da20e033fe94f8396493761d20c5aef14441aa0c9f681");
INSERT INTO Hebergeur VALUES (2, "Henri CASTAFOLTE", "3 souterrain du passant", "+33112357894",
"henri.castafolte@vdf.fr", "3a3e0ae1fe27735a8db6ae0060af663c66dc8d5295828e1206b34b74d50bb5bf");
```



```

INSERT INTO Hebergeur VALUES (3, "Danielle ROUSSEL", "33 boulevard de Normandie",
"+33519806914", "danielleroussel@jourrapide.com",
"07567d11b04ae73a2250b27c94ae49146818e07d8bd20bfdc9f7afba7186c579");
INSERT INTO Hebergeur VALUES (4, "Rosamonde BELLEFEUILLE", "2 rue Charles Corbeau",
"+33217754232", "rosamondebellefeuille@rhyta.com",
"2f72fa9fb3487ffbaf654cedfel3e9ddee0f24e3a7e453a15b858463cccd515");
INSERT INTO Hebergeur VALUES (5, "Chantal BAZINET", "17 rue du General Ailleret",
"+33235385349", "chantalbazinet@rhyta.com",
"ec98cdff032fbc2c2eb1d3edfd66636477c1ebe488de7985ff64984e7cda7568");
INSERT INTO Hebergeur VALUES (6, "Arnaude THIVI", "30 Square de la Couronne", "+33124029297",
"arnaudethivierge@jourrapide.com",
"44d9c62ab336ee955a7c560507792f0b5089861fec4c9840c74a9d29fddc2e72");

INSERT INTO Client VALUES (1, "Wayne FERGUSON", "90 rue de la Mare aux Carats", "+33155597536",
"wayne.ferguson@hotmail.com",
"6b86b273ff34fcel9d6b804eff5a3f5747ada4eaa22f1d49c01e52ddb7875b4b", "1988-10-14", "M", "fr");
INSERT INTO Client VALUES (2, "Hu LEBLANC", "59 rue de la Fosse des Tanneurs", "+33429558678",
"leblanchu@hotmail.edu", "d4735e3a265e16eee03f59718b9b5d03019c07d8b6c51f90da3a666eec13ab35",
"1975-10-14", "M", "fr");
INSERT INTO Client VALUES (3, "Gil CROMWELL", "59 rue Bonneterie", "+33471443947",
"gil.chmwill@gmail.com", "4e07408562bedb8b60ce05c1decfe3ad16b72230967de01f640b7e4729b49fce",
"1980-12-02", "M", "us");
INSERT INTO Client VALUES (4, "Odysseus AAAA", "29 avenue Voltaire", "+33873781894",
"aaaa@protonmail.com", "4b22777d4dd1fc61c6f884f48641d02b4d121d3fd328cb08b5531fcacdabf8a",
"1998-05-02", "F", "fr");
INSERT INTO Client VALUES (5, "Aurore LYNN", "37 avenue des pres", "+33556180844",
"aurore@outlook.net", "ef2d127de37b942baad06145e54b0c619a1f22327b2ebbcfbec78f5564afe39d", "1974-
10-18", "F", "fr");
INSERT INTO Client VALUES (6, "Madonna BRIEN", "76 place Napoleon", "+33255617743",
"themadd@protonmail.ca", "e7f6c011776e8db7cd330b54174fd76f7d0216b612387a5ffcfb81e6f0919683",
"1990-12-27", "F", "us");
INSERT INTO Client VALUES (7, "Vera JEUSEPA", "66 boulevard Amiral Courbet", "+33233748258",
"jeusepaki@gmail.com", "7902699be42c8a8e46fbbb4501726517e86b22c56a189f7625a6da49081b2451",
"1989-08-31", "F", "fr");
INSERT INTO Client VALUES (8, "Mara RIOS", "62 Cours Marechal-Joffre", "+33962492263",
"maria.rios@yahoo.couk", "2c624232cdd221771294dfbb310aca000a0df6ac8b66b696d90ef06fdefb64a3",
"1979-12-30", "M", "es");
INSERT INTO Client VALUES (9, "Ciaran PONTS", "74 rue du Faubourg National", "+33452514178",
"pontsu@aol.com", "19581e27de7ced00ff1ce50b2047e7a567c76b1cbaebabe5ef03f7c3017bb5b7", "1974-09-
29", "M", "fr");
INSERT INTO Client VALUES (10, "Barclay EEEE", "42 rue Bonnet", "+33225152858",
"eeee@icloud.couk", "4a44dc15364204a80fe80e9039455cc1608281820fe2b24f1e5233ade6af1dd5", "1977-
08-23", "F", "uk");

INSERT INTO Hebergement VALUES (1, 1, "APPMaison", 1);
INSERT INTO Hebergement VALUES (2, 2, "APPMaison", 1);
INSERT INTO Hebergement VALUES (3, 3, "APPMaison", 1);
INSERT INTO Hebergement VALUES (4, 1, "HOTEL", 1);
INSERT INTO Hebergement VALUES (7, 2, "HOTEL", 2);
INSERT INTO Hebergement VALUES (5, 1, "APPHOTEL", 1);
INSERT INTO Hebergement VALUES (6, 2, "APPHOTEL", 1);

INSERT INTO ReservationHebergement VALUES (1, 1, "2023-02-25", "2023-02-26", 1, 0, "Marzy",
"France", 0);
INSERT INTO ReservationVol VALUES (2, "FRJP 0000 MNGA 2109", "2023-03-12", "2023-03-12", 2, 0,
0, 0);
INSERT INTO ReservationSejour VALUES (3, 7, "FRUS 0000 CPP4 0801", "Formule IDP");

```

Scripts de manipulation

InscrireClient.sql

```
SET @Nom_Prenom = "Jean DUPONT";
SET @Adresse = "5 rue des oies";
SET @Num_Tel = "+33736211227";
SET @Email = "jean.dupont@gmail.com";
SET @MDP = "5994471abb01112afcc18159f6cc74b4f511b99806da59b3caf5a9c173cacfc5"; -- Sous la forme
d'un hash SHA-256
SET @DateNaissance = "1865-04-05";
SET @Sexe = "M";
SET @Langue = "fr"; -- Sous la forme d'un code ISO 639-1

INSERT INTO Client VALUES (0, @Nom_Prenom, @Adresse, @Num_Tel, @Email, @MDP, @DateNaissance,
@Sexe, @Langue);
```

InscrireHebergeur.sql

```
SET @Nom_Prenom = "Florian DELAVERNHE";
SET @Adresse = "2 rue du petit bois";
SET @Num_Tel = "+33789632514";
SET @Email = "darkflofloangers@gmail.com";
SET @MDP = "5994471abb01112afcc18159f6cc74b4f511b99806da89b3caf5a9c174cacfc2"; -- Sous la forme
d'un hash SHA-256

INSERT INTO Hebergeur VALUES (0, @Nom_Prenom, @Adresse, @Num_Tel, @Email, @MDP);
```

CreeAvion.sql

```
SET @Type = "Airbus A-330";
SET @Nb_Sieges = 350;

INSERT INTO Avion VALUES (0, @Type, @Nb_Sieges);
```

CreeSiege.sql

```
SET @Nb_Avion = 0;
SET @Prem_Classe = 0;
SET @Cote_Hublot = 0;

INSERT INTO Siege VALUES (0, @Nb_Avion, @Prem_Classe, @Cote_Hublot);
```

CreeVol.sql

```
SET @Nb_Avion = 0;
SET @Identifiant_Vol = "0000 0000 0000 0000";
SET @Aero_Depart = "Orly";
SET @Aero_Arrive = "Felix Eboue";
SET @Horaire_Arrive = "14:00:00";
SET @Horaire_Depart = "08:00:00";
SET @Tarif_Eco = 250.0; -- En Euros (€)
SET @Tarif_Eco_Hublot = 350.0; -- En Euros (€)
SET @Tarif_Prem = 400.0; -- En Euros (€)
SET @Tarif_Prem_Hublot = 450.0; -- En Euros (€)
SET @Bagages_Inclus = 0;
SET @Tarif_Ajout_1_Bagage = 50.0; -- En Euros (€)
SET @Tarif_Ajout_2_Bagages = 100.0; -- En Euros (€)
SET @Tarif_Ajout_3_Bagages = 150.0; -- En Euros (€)

INSERT INTO Vol VALUES (@Nb_Avion, @Identifiant_Vol, @Aero_Depart, @Aero_Arrive,
@Horaire_Arrive, @Horaire_Depart, @Tarif_Eco, @Tarif_Eco_Hublot, @Tarif_Prem,
@Tarif_Prem_Hublot, @Bagages_Inclus, @Tarif_Ajout_1_Bagage, @Tarif_Ajout_2_Bagages,
@Tarif_Ajout_3_Bagages);
```

ReserverVol.sql

```
SET @Identifiant_Vol = "0000 0000 0000 0000";
SET @Date_Depart = "2023-10-04";
SET @Date_Arrive = "2023-10-04";
SET @Nb_Voyageurs = 1;
SET @Aller_Retour = 0;
SET @Prem_Classe = 0;
SET @Avec_Escale = 0;

INSERT INTO ReservationVol VALUES (0, @Identifiant_Vol, @Date_Depart, @Date_Arrive,
@Nb_Voyageurs, @Aller_Retour, @Prem_Classe, @Avec_Escale);
```

ReserverHebergement.sql

```
SET @Nb_Hebergement = 1;
SET @Date_Arrivee = "2023-10-04";
SET @Date_Depart = "2023-10-04";
SET @Nb_Adultes = 1;
SET @Nb_Enfants = 2;
SET @Ville = "Paris";
SET @Pays = "France";
SET @Remboursable = 1;

INSERT INTO Hebergement VALUES (0, @Nb_Hebergement, @Date_Arrivee, @Date_Depart, @Nb_Adultes,
@Nb_Enfants, @Ville, @Pays, @Remboursable);
```

ReserverSejour.sql

```
SET @Nb_Client = 2;
SET @Nb_Unique_Hbgmt = 8;
SET @Identifiant_Vol = "0000 0000 0000 0000";
SET @Formule = "Formule EXEMPLE";

INSERT INTO ReservationSejour VALUES (@Nb_Client, @Nb_Unique_Hbgmt, @Identifiant_Vol, @Formule);
```

CreeAppartMaison.sql

```
SET @Nb_Hebergeur = 1;

SET @Nom = "Maison de la Bellevue";
SET @Pays = "France";
SET @Adresse = "47 chemin de la Riviere";
SET @Emplacement_Hebergeables = 2;
SET @Nb_Salle_De_Bain = 1;
SET @Superficie = 20;
SET @Equipement = "Deux lits, Une SDB avec douche, Toilettes, Lavabo, Television, Volets electriques";
SET @Parking = 1;
SET @Parking_Payant = 0;
SET @Fumeur = 0;
SET @Animaux = 0;
SET @Enfants = 1;
SET @Fetes = 0;
SET @Commentaire = "aucun commentaire";
SET @Noms_Photos = "exemple1.png|exemple2.png";

INSERT INTO AppartementMaison VALUES (0, @Nom, @Pays, @Adresse, @Emplacement_Hebergeables, @Nb_Salle_De_Bain, @Superficie, @Equipement, @Parking, @Parking_Payant, @Fumeur, @Animaux, @Enfants, @Fetes, @Commentaire, @Noms_Photos);
INSERT INTO Hebergement VALUES (0, (SELECT nb_appmaison FROM AppartementMaison ORDER BY nb_appmaison DESC LIMIT 1), "APPMaison", @Nb_Hebergeur);
```

CreeHotel.sql

```
SET @Nb_Hebergeur = 1;

SET @Nom = "Hotel de la mer";
SET @Localite = "Paris";
SET @Adresse = "10 rue de l'Hotel de ville";
SET @Nb_Etoile = 2;
SET @Nom_Contact = "Maxime BERGER";
SET @Num_Telephone = "+33725658596";
SET @Num_Tel2 = "+33386542598";
SET @Chaine_Hoteliere = "IBIS";
SET @Equipement = "Lit deux places, SDB, Douche, Toilettes, Television";
SET @Horaire_Arrivee = "12:00:00";
SET @Horaire_Depart = "10:00:00";
SET @Fumeur = 0;
SET @Animaux = 0;
SET @Enfant = 0;
SET @Fete = 0;
SET @Ptit_Dej = 1;
SET @Langues = "fr";
SET @Navette_Aeroport = 0;
SET @Resto_Sur_Place = 1;
SET @Formule_Resto = "Entree, Plat, Fromage ou dessert";
SET @Lits_Appoint = 1;
SET @Tarif_Lit_Appoint = 10.50;

INSERT INTO Hotel VALUES (0, @Nom, @Localite, @Adresse, @Nb_Etoile, @Nom_Contact, @Num_Telephone, @Num_Tel2, @Chaine_Hoteliere, @Equipement, @Horaire_Arrivee, @Horaire_Depart, @Fumeur, @Animaux, @Enfant, @Fete, @Ptit_Dej, @Langues, @Navette_Aeroport, @Resto_Sur_Place, @Formule_Resto, @Lits_Appoint, @Tarif_Lit_Appoint);
INSERT INTO Hebergement VALUES (0, (SELECT nb_hotel FROM Hotel ORDER BY nb_hotel DESC LIMIT 1), "HOTEL", @Nb_Hebergeur);
```

CreeChambreHotel.sql

```
SET @Nb_Hotel = 10;
SET @Type_Chambre = "Chambre 1 Personne";
SET @Superficie = 15; -- En m²
SET @Vue = "Vue sur cour";
SET @Tarif = 65.50; -- En Euros (€)
SET @Equipement = "Douche, Toiletttes, Lavabo, Television";

INSERT INTO Chambre VALUES (0, @Nb_Hotel, @Type_Chambre, @Superficie, @Vue, @Tarif,
@Equipement);
```

CreeAppartHotel.sql

```
SET @Nb_Hebergeur = 1;

SET @Nom = "Appart-Hotel du saint donut";
SET @Localite = "Nevers";
SET @Adresse = "9 rue d'auvergne";
SET @Equipements = "Television, Baignoire";
SET @Horaire_Arrivee = "10:00:00";
SET @Horaire_Depart = "08:00:00";
SET @Fumeur = 0;
SET @Animaux = 0;
SET @Enfant = 1;
SET @Fete = 0;
SET @Ptit_Dej = 0;
SET @Langues = "fr";
SET @Douche_ou_Baignoire = 1;
SET @Serviettes = 1;
SET @Seche_cheveux = 0;
SET @Micro_ondes = 0;
SET @Plaques_cuisson = 1;
SET @Ustensiles_cuisine = 1;
SET @Bouilloire = 0;
SET @Frigo = 1;
SET @Tarif = 65.99; -- En Euros (€)

INSERT INTO AppartHotel VALUES (0, @Nom, @Localite, @Adresse, @Equipements, @Horaire_Arrivee,
@Horaire_Depart, @Fumeur, @Animaux, @Enfant, @Fete, @Ptit_Dej, @Langues, @Douche_ou_Baignoire,
@Serviettes, @Seche_cheveux, @Micro_ondes, @Plaques_cuisson, @Ustensiles_cuisine, @Bouilloire,
@Frigo, @Tarif);
INSERT INTO Hebergement VALUES (0, (SELECT nb_apphotel FROM AppartHotel ORDER BY nb_apphotel
DESC LIMIT 1), "APPHOTEL", @Nb_Hebergeur);
```

RelieAppartAAppartHotel.sql

```
SET @Nb_Hotel = 1;

SET @Emplacement_Hebergeables = "Chambre";
SET @Nb_Personnes_Hebergeables = 2;
SET @Nb_SDB = 1;
SET @Superficie = 30; -- En m²
SET @Nb_Chambre = 1;
SET @Noms_Photos = "apphotel-app1.png|apphotel-app2.png";

INSERT INTO HotelAppart VALUES (0, @Emplacement_Hebergeables, @Nb_Personnes_Hebergeables,
@Nb_Salle_De_Bain, @Superficie, @Nb_Chambre, @Noms_Photos);
INSERT INTO COMPRENDSAPPART VALUES ((SELECT nb_appart FROM HotelAppart ORDER BY nb_appart DESC
LIMIT 1), @Nb_Hotel);
```

RelieStudioAppartHotel.sql

```
SET @Nb_Hotel = 1;

SET @Nb_Lits_simples = 1;
SET @Nb_Lits_doubles = 0;
SET @Nb_Personnes = 1;
SET @Superficie = 30; -- En m²
SET @Noms_Photos = "apphotel-studio1.png|apphotel-studio1.png";

INSERT INTO HotelStudio VALUES (0, @Nb_Lits_simples, @Nb_Lits_doubles, @Nb_Personnes,
@Superficie, @Noms_Photos);
INSERT INTO COMPRENDSSSTUDIO VALUES ((SELECT nb_studio FROM HotelStudio ORDER BY nb_studio DESC
LIMIT 1), @Nb_Hotel);
```

VoirAppartMaisonDeHebergeur.sql

```
SET @Nom_Hebergeur = "Monique DUPONT";

SELECT * FROM AppartementMaison WHERE nb_appmaison = ANY(SELECT nb_hebergement FROM Hebergement
WHERE type = "APPMAISON" AND nb_hebergeur = ANY(SELECT nb_hebergeur FROM Hebergeur WHERE
nom_prenom = @Nom_Hebergeur));
```

VoirHotelsDeHebergeur.sql

```
SET @Nom_Hebergeur = "Monique DUPONT";

SELECT * FROM Hotel WHERE nb_hotel = ANY(SELECT nb_hebergement FROM Hebergement WHERE type =
"Hotel" AND nb_hebergeur = ANY(SELECT nb_hebergeur FROM Hebergeur WHERE nom_prenom =
@Nom_Hebergeur));
```

VoirChambresDeHotel.sql

```
SET @Nb_Hotel = 1;

SELECT * FROM Chambre WHERE nb_hotel = @Nb_Hotel;
```

VoirAppartHotelDeHebergeur.sql

```
SET @Nom_Hebergeur = "Monique DUPONT";

SELECT * FROM AppartHotel WHERE nb_apphotel = ANY(SELECT nb_hebergement FROM Hebergement WHERE
type = "APPHOTEL" AND nb_hebergeur = ANY(SELECT nb_hebergeur FROM Hebergeur WHERE nom_prenom =
@Nom_Hebergeur));
```

VoirClient.sql

```
SET @Nom_Client = "Wayne FERGUSON";

SELECT * FROM Client WHERE nom_prenom = @Nom_Client;
```

VoirHebergeur.sql

```
SET @Nom_Hebergeur = "Monique DUPONT";

SELECT * FROM Hebergeur WHERE nom_prenom = @Nom_Hebergeur;
```