

# RAPPORT DE MINI PROJET

Agence de Voyage

PRÉSENTÉ À

Mohammed Nahbaoui

PRÉSENTÉ PAR

El Mahdi Bouaiti

# SOMMAIRE

Introduction.....	3
1-Cahier de charge.....	4
2-Règles des gestion.....	5
3-Dictionnaire de Données.....	7
4-Model Conceptuelle de donnée.....	10
5-Model Logique de donnée.....	13
6-Creation de la base de donnée.....	15
7-Remplissage de la base de donnée.....	17
8-Test de la base de donnée.....	24
8-Test de la base de donnée.....	26

# Introduction

De nos jours, l'informatique est un outil indispensable pour les chefs d'entreprise qui veulent rester compétitifs dans un monde en constante évolution technologique. Ainsi, dans le cadre d'un mini-projet de deuxième année en génie informatique portant sur la base de données MySQL, M. Nahbaoui souhaite créer un système de travail pour développer une application de gestion d'une agence de voyages. L'objectif est de concevoir une base de données pour une agence de voyages qui offre différents circuits. Grâce à cette base de données, les utilisateurs pourront visualiser les circuits proposés par l'agence, consulter leurs descriptions et réserver leur voyage. L'application aura également une interface d'administration qui permettra à l'administrateur connecté de visualiser, ajouter et supprimer les voyages proposés par l'agence. Pour mener à bien ce projet, différentes tâches ont été réparties, telles que la réflexion sur le fonctionnement de la base de données et les relations entre les différentes parties, ainsi que la création d'un schéma entités-associations correspondant à la base de données. Ce cahier des charges répond aux besoins et exigences exprimés par le client et vise à créer une application permettant à une agence de voyages de gérer la vente de leurs offres de voyage et de permettre aux clients de visualiser toutes les offres pour qu'ils puissent choisir leur voyage.

# 1-Cahier de charges

## Coté agence

- Gestion des voyages
- Gestion des programmations des voyages
- Consultation des voyages (par catégorie, date, ...)
- Gestion des points de départ des voyages (plusieurs point par voyage)
- Gestion des autocars et de leur affectation à une programmation
- Gestion des hôtels et de leur affectation à un voyage (un hotel par voyage)
- Gestion du nombre de places et des emplacements dans les autocars et des types d'autocars
- Gestion des prix en fonction de la période
- Gestion des points de départ
- Coté client
- Inscription avec coordonnées personnelles
- Consultation des voyages par catégorie, date, destination, ...
- Réserver un voyage pour lui-même et/ou d' autres personnes
- Choisir son emplacement dans l'autocar
- Choisir un point de départ pour le voyage
- Choisir des suppléments (Assurance annulation, chambre supplémentaire, ...)

## 2-Regles de gestion

### Autocars

- Il existe plusieurs types d'autocar
- Un Autocar appartient à un et un seul type d'autocar
- Un emplacement appartient à un et un seul autocar

### Passagers

- Un passager possède un et un seul emplacement et une et une seule réservation
- Le passager doit impérativement avec des coordonnées

### Réservations

- Une réservation peut concerner un ou plusieurs passagers
- Une assurance annulation peut être prise pour une réservation
- Une chambre supplémentaire peut être prise pour une réservation
- Une réservation est faite par un et un seul client

### Clients

- Un client peut effectuer aucune ou plusieurs réservations
- Un client doit avoir un genre
- Un client habite une et une seule ville

## 2-Regles de gestion

### Villes / Départements / Région / Pays

- Une ville est dans un et un seul département
- Un département est dans une et une seule région
- Une région est dans un et un seul pays
- Une ville peut avoir un ou plusieurs points de départ
- Une ville peut avoir aucun ou plusieurs Hôtels

### Voyages

- Un voyage se voit affecter un et un seul hôtel
- Un voyage est soit en pension complète soit en demi-pension mais le choix n'est pas donné
- Un voyage peut être d'un et un seul type (circuit, séjour, ...)
- Un voyage peut être programmé plusieurs fois dans l'année

### Programmations

- Une programmation concerne un et un seul voyage
- Une programmation peut se voir affecter un ou plusieurs Autocars
- Le prix de base de la programmation est à renseigner manuellement par l'utilisateur via l'interface de gestion des programmations.

### Hôtels

- Un hôtel peut être affecté à 0 ou plusieurs voyages

## 3-Dictionnaire de Données

Le dictionnaire de données décrit les entités et les attributs d'une base de données pour la gestion de Agence de voyage. Il comprend des informations sur les autocars tels que leur ID, leur marque, leur modèle et le nombre de places. Il contient également des données sur les passagers tels que leur ID, leur nom, leur prénom, leur adresse, leur numéro de téléphone et leur adresse e-mail. Les réservations sont liées aux voyages et aux clients, avec des attributs tels que l'ID de la réservation, l'ID du voyage, l'ID du client, la date de réservation et le nombre de places réservées. Enfin, la base de données contient des informations géographiques sur les villes, les départements, les régions et les pays, ainsi que des données sur les voyages, les hôtels et les programmations prévues.

Le tableau ci-dessous présente le dictionnaire de données pour la gestion de réservations d'autocars.

## 3-Dictionnaire de Données

Code mnémonique	Désignation	Type	Taille	Remarque
Entité: Autocars				
ID_AUT	Id de l'autocar	Entier	8	Clé primaire
MARQUE	Marque	Texte	50	
MODELE	Modèle	Texte	50	
NB_PLACES	Nombre de places	Entier	8	
Entité: Passagers				
ID_PASS	Id du passager	Entier	8	Clé primaire
NOM_PASS	Nom	Texte	50	
PRENOM_PASS	Prénom	Texte	50	
ADRESSE_PASS	Adresse	Texte	100	
TEL	Numéro de téléphone	Entier	20	
EMAIL_PASS	Adresse Email	texte	50	
Entité: Réservations				
ID_RES	Id de la réservation	Entier	8	Clé primaire
ID_VOY	Id du voyage	Entier	8	Clé étrangère
ID_CLI	Id du client	Entier	8	Clé étrangère
DATE_RES	Date de réservation	Date/H eure		
NB_RES	Nombre de places réservées	Entier	8	
Entité: Clients				
ID_CLIENT	Id du client	Entier	8	Clé primaire
NOM_CLIENT	Nom	Texte	50	
PRENOM_CLIENT	Prénom	Texte	50	
ADRESSE_CLIENT	Adresse	Texte	100	
TEL_CLIENT	Numéro de téléphone	Entier	20	
EMAIL_CLIENT	Adresse Email	Texte	50	

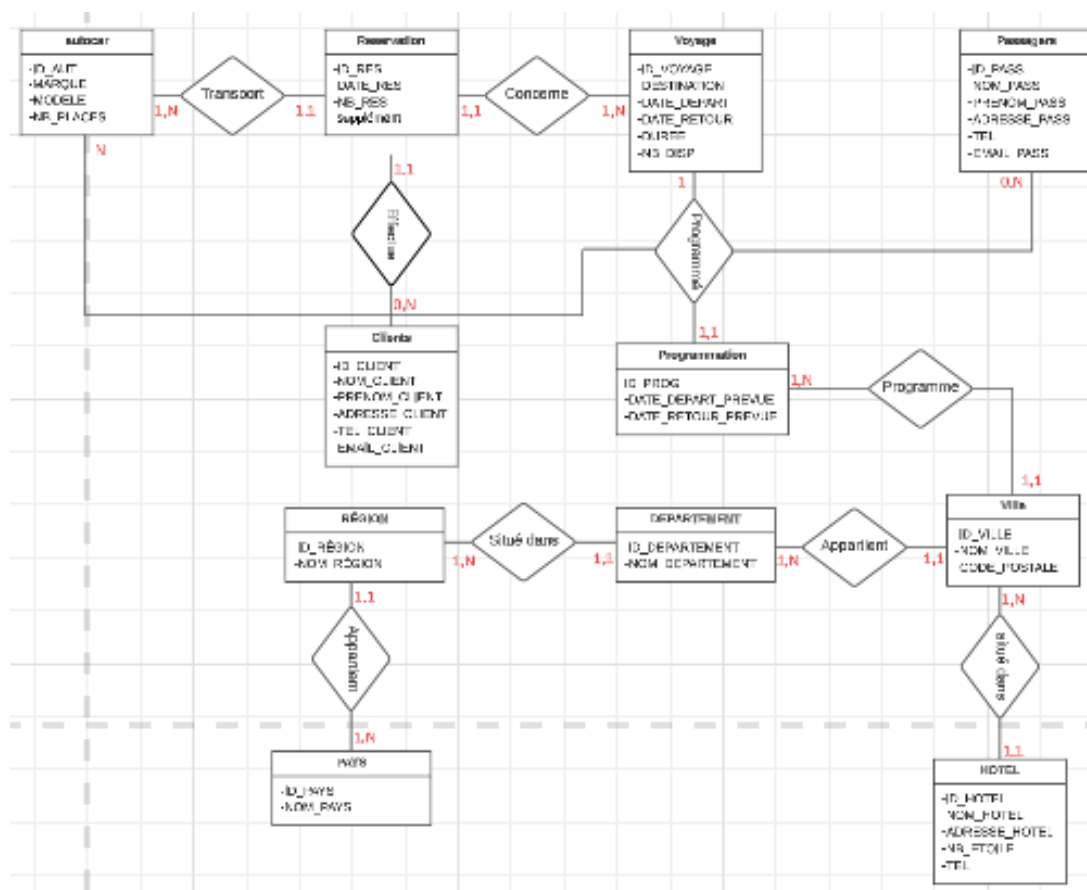


## 3-Dictionnaire de Données

Entité: Ville/département/région/pays				
ID_VILLE	Id de la ville/département/région/pays	Entier	8	Clé primaire
NOM_VILLE	Nom	Texte	50	
DEPARTEMENT	Département	Texte	50	Si applicable
REGION	Région	Texte	50	Si applicable
PAYS	Pays	Texte	50	
CODE_POSTALE	Code postale de la ville	Entier	8	
Entité: Voyage				
ID_VOYAGE	Id du voyage	Entier	8	Clé primaire
DESTINATION	Destination	Texte	50	Clé étrangère
DATE_DEPART	Date de départ	Date/Heure		
DATE_RETOUR	Date de retour	Date/Heure		
DUREE	Durée	Entier	8	
NB_DISP	Nombre de places disponibles	Entier	8	
Entité: Programmation				
ID_PROG	Id de la programmation	Entier	8	Clé primaire
DATE_DEPART_PREVUE	Date de départ prévue	Date/Heure		
DATE_RETOUR_PREVUE	Date de retour prévue	Date/Heure		
Entité: Hôtel				
ID_HOTEL	Id de l'hôtel	Entier	8	Clé primaire
NOM_HOTEL	Nom de l'hôtel	Texte	100	
ADRESSE_HOTEL	Adresse de l'hôtel	Texte	100	
NB_ETOILE	Nombre des étoiles	Entier	5	
TEL	Numéro de téléphone	Entier	20	

## 4-Model Conceptuelle de donnée

La modélisation conceptuelle des données joue un rôle essentiel dans la compréhension et la structuration des différents éléments de notre système. Elle nous permet de visualiser les relations entre les entités et de définir les attributs qui les caractérisent. En établissant un modèle conceptuel solide, nous facilitons la création d'une base de données efficace et cohérente. Dans cette optique, nous avons développé un modèle conceptuel de données qui décrit les différentes entités et leurs interactions au sein de notre système de réservation de voyages. Examinons de plus près les détails de ce modèle.



## 4-Model Conceptuelle de donnée

Les clients, représentant les personnes inscrites sur le site, ont la possibilité de réaliser des réservations. Il est important de souligner que tous les clients ne sont pas nécessairement des passagers. En effet, un client peut effectuer une réservation sans être un passager, et inversement, un passager peut ne pas être un client.

La partie essentielle de notre système concerne les voyages. Elle englobe la destination, un ou plusieurs points de départ, ainsi qu'un choix de pension (complète ou demi-pension). Le prix du voyage varie en fonction de la période de l'année. Il est crucial de noter que lorsqu'un client effectue une réservation, celle-ci est liée à une programmation spécifique, qui correspond à un voyage planifié à une date donnée.

La table des passagers confirme ce que nous avons précédemment mentionné à propos des clients. En effet, les passagers ne sont pas les individus ayant réalisé la réservation, mais plutôt ceux qui participent à la programmation du voyage.

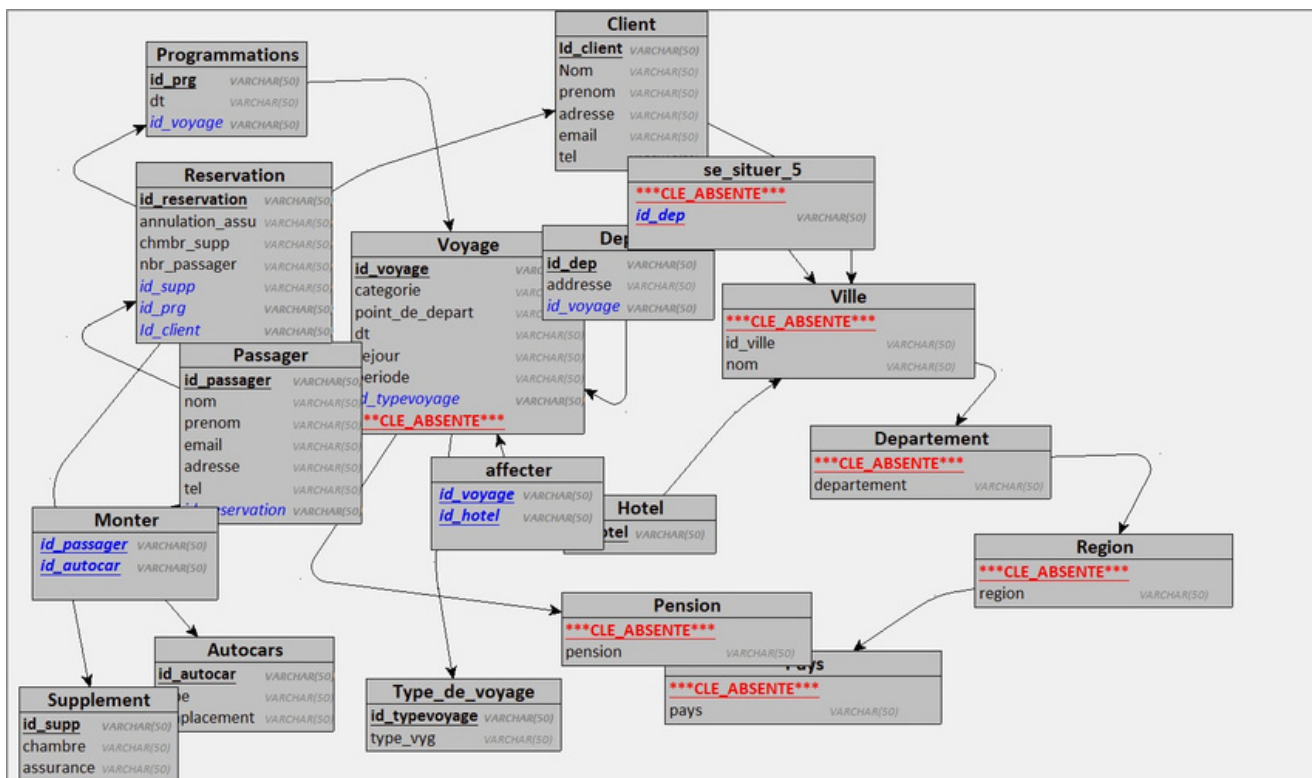
Lorsqu'un client effectue une réservation, il peut éventuellement choisir des suppléments tels qu'une chambre individuelle ou une assurance annulation.

## 4-Model Conceptuelle de donnée

Enfin, il est nécessaire d'attribuer un ou plusieurs autocars à une programmation spécifique. Ces autocars sont catégorisés en différents types, ce qui nous permet d'associer à chacun un plan d'autocar et un nombre maximal de places. Chaque passager se voit attribuer une place précise dans un autocar donné. Ainsi, un client est en mesure de réserver une ou plusieurs places spécifiques en fonction des autocars assignés, des plans correspondants et des places restantes disponibles.

## 5-Model Logique de donnée

Le modèle logique de données représente une étape clé dans la conception de notre système de réservation de voyages. Il traduit le modèle conceptuel en une structure plus concrète, décrivant les tables, les attributs et les relations entre les entités.



Le modèle logique se compose de plusieurs entités, dont les principales sont les clients, les voyages, les programmations, les passagers et les autocars. La table des clients enregistre les informations sur les personnes inscrites sur le site, telles que leurs noms, coordonnées et préférences. Les voyages sont représentés par une table comprenant des détails tels que la destination, les points de départ et le type de pension.

## 5-Model Logique de donnée

La table des programmations est liée aux voyages et permet de planifier les différentes dates auxquelles les voyages auront lieu. Chaque programmation est associée à un voyage spécifique. Les passagers sont enregistrés dans une table distincte, contenant des informations telles que leurs noms, leurs dates de naissance et leurs préférences de siège.

Pour associer les clients, les programmations et les passagers, nous utilisons des clés étrangères. Par exemple, un client peut effectuer une réservation pour une programmation spécifique en utilisant la clé étrangère qui lie la table des clients à celle des programmations. De même, la table des passagers est liée à la table des programmations pour indiquer quels passagers participent à quelles programmations.

En ce qui concerne les autocars, ils sont représentés par une table distincte, avec des informations sur les types d'autocars, les plans d'autocar associés et le nombre maximal de places disponibles. Chaque programmation est ensuite liée à un ou plusieurs autocars en utilisant des clés étrangères.

En résumé, le modèle logique de données définit la structure et les relations des tables qui constituent notre système de réservation de voyages. Il permet de capturer les différentes entités impliquées, leurs attributs et leurs interactions, facilitant ainsi la mise en œuvre d'une base de données efficace et cohérente.

## 6-Creation de la base de donnée

```

/* /* Creation de la base de données*/
CREATE DATABASE Agence_Voyage;
/* utilisation de la base de données*/
use Agence_Voyage;
/* Creation de la table autocar*/
CREATE TABLE Autocars (
    id_autocar INT AUTO_INCREMENT,
    type VARCHAR(50),
    emplacement VARCHAR(50),
    PRIMARY KEY (id_autocar)
) ENGINE=InnoDB;
/* Creation de la table pays*/
CREATE TABLE Pays (
    id_pays INT AUTO_INCREMENT,
    pays VARCHAR(50),
    PRIMARY KEY (id_pays)
) ENGINE=InnoDB;
/* Creation de la table pension*/
CREATE TABLE Pension (
    id_pension INT AUTO_INCREMENT,
    pension VARCHAR(50),
    PRIMARY KEY (id_pension)
) ENGINE=InnoDB;
/* Creation de la table type de voyage*/
CREATE TABLE Type_de_voyage (
    id_typevoyage VARCHAR(50),
    type_vyg VARCHAR(50),
    PRIMARY KEY (id_typevoyage)
) ENGINE=InnoDB;
/* Creation de la table supplement*/
CREATE TABLE Supplement (
    id_supp INT AUTO_INCREMENT,
    chambre VARCHAR(50),
    assurance VARCHAR(50),
    PRIMARY KEY (id_supp)
) ENGINE=InnoDB;

```

```

/* Creation de la table voyage*/
CREATE TABLE Voyage (
    id_voyage INT AUTO_INCREMENT,
    categorie VARCHAR(50),
    point_de_depart VARCHAR(50),
    dt VARCHAR(50),
    sejour VARCHAR(50),
    periode VARCHAR(50),
    id_typevoyage VARCHAR(50),
    id_pays INT,
    PRIMARY KEY (id_voyage),
    FOREIGN KEY (id_typevoyage) REFERENCES Type_de_voyage(id_typevoyage),
    FOREIGN KEY (id_pays) REFERENCES Pays(id_pays)
) ENGINE=InnoDB;
/* Creation de la table programmation*/
CREATE TABLE Programmmations (
    id_prg INT AUTO_INCREMENT,
    dt VARCHAR(50),
    id_voyage INT,
    PRIMARY KEY (id_prg),
    FOREIGN KEY (id_voyage) REFERENCES Voyage(id_voyage)
) ENGINE=InnoDB;
/* Creation de la table region*/
CREATE TABLE Region (
    id_region INT AUTO_INCREMENT,
    region VARCHAR(50),
    PRIMARY KEY (id_region)
) ENGINE=InnoDB;
/* Creation de la table depart*/
CREATE TABLE Depart (
    id_dep INT AUTO_INCREMENT,
    adresse VARCHAR(50),
    id_voyage INT,
    PRIMARY KEY (id_dep),
    FOREIGN KEY (id_voyage) REFERENCES Voyage(id_voyage)
) ENGINE=InnoDB;

```

## 6-Creation de la base de donnée

```
/* Creation de la table departement*/
CREATE TABLE Departement (
    id_dept INT AUTO_INCREMENT,
    departement VARCHAR(50),
    PRIMARY KEY (id_dept)
) ENGINE=InnoDB;

/* Creation de la table ville*/
CREATE TABLE Ville (
    id_ville INT AUTO_INCREMENT,
    id_voyage INT,
    nom VARCHAR(50),
    PRIMARY KEY (id_ville),
    FOREIGN KEY (id_voyage) REFERENCES Voyage(id_voyage)
) ENGINE=InnoDB;

/* Creation de la table client*/
CREATE TABLE Client (
    id_client INT AUTO_INCREMENT,
    Nom VARCHAR(50),
    prenom VARCHAR(50),
    adresse VARCHAR(50),
    email VARCHAR(50),
    tel VARCHAR(50),
    PRIMARY KEY (id_client)
) ENGINE=InnoDB;

/* Creation de la table hotel*/
CREATE TABLE Hotel (
    id_hotel INT AUTO_INCREMENT,
    nom_hotel VARCHAR(50),
    PRIMARY KEY (id_hotel)
) ENGINE=InnoDB;
```



## 6-Creation de la base de donnée

```
/* Creation de la table reservation*/
CREATE TABLE Reservation (
  id_reservation INT AUTO_INCREMENT,
  annulation_assu VARCHAR(50),
  chmbr_supp VARCHAR(50),
  nbr_passager VARCHAR(50),
  id_supp INT,
  id_prg INT,
  id_client INT,
  PRIMARY KEY (id_reservation),
  FOREIGN KEY (id_supp) REFERENCES Supplement(id_supp),
  FOREIGN KEY (id_prg) REFERENCES Programmmations(id_prg),
  FOREIGN KEY (id_client) REFERENCES Client(id_client)
) ENGINE=InnoDB;

/* Creation de la table passager*/
CREATE TABLE Passager (
  id_passager INT AUTO_INCREMENT,
  nom VARCHAR(50),
  prenom VARCHAR(50),
  adresse VARCHAR(50),
  email VARCHAR(50),
  tel VARCHAR(50),
  id_reservation INT,
  PRIMARY KEY (id_passager),
  FOREIGN KEY (id_reservation) REFERENCES Reservation(id_reservation)
) ENGINE=InnoDB;

/* Creation de la table affecter*/
CREATE TABLE affecter (
  id_voyage INT,
  id_hotel INT,
  FOREIGN KEY (id_voyage) REFERENCES Voyage(id_voyage),
  FOREIGN KEY (id_hotel) REFERENCES Hotel(id_hotel)
) ENGINE=InnoDB;

/* Creation de la table Monter*/
CREATE TABLE Monter (
  id_passager INT,
  id_autocar INT,
  FOREIGN KEY (id_passager) REFERENCES Passager(id_passager),
  FOREIGN KEY (id_autocar) REFERENCES Autocars(id_autocar)
) ENGINE=InnoDB;
```

## 7-Remplissage de la base de donnée

```
/* Inserting data into Pension table */
INSERT INTO Pension (pension)
VALUES
('All-inclusive'),
('Half-board'),
('Full-board'),
('Bed and breakfast'),
('Room only'),
('Self-catering'),
('Ultra All-inclusive'),
('Cruise'),
('No meals'),
('Breakfast only');

/* Inserting data into Type_de_voyage table */
INSERT INTO Type_de_voyage (id_typevoyage, type_vyg)
VALUES
('SEA', 'Sea'),
('SUN', 'Sun'),
('CTY', 'City'),
('DIS', 'Discovery'),
('MOT', 'Mountain'),
('SPA', 'Spa'),
('HON', 'Honeymoon'),
('ADV', 'Adventure'),
('CUL', 'Cultural'),
('FAM', 'Family');
```

## 7-Remplissage de la base de donnée

```
/* Inserting data into Autocars table */
INSERT INTO Autocars (type, emplacement)
VALUES
    ('Mercedes-Benz', 50),
    ('Iveco', 40),
    ('Volvo', 55),
    ('MAN', 48),
    ('Scania', 52),
    ('Setra', 45),
    ('Renault', 38),
    ('Neoplan', 42),
    ('Van Hool', 50),
    ('Temsa', 44);

/* Inserting data into Pays table */
INSERT INTO Pays (pays)
VALUES
    ('France'),
    ('Spain'),
    ('Italy'),
    ('Portugal'),
    ('Germany'),
    ('Greece'),
    ('Croatia'),
    ('Belgium'),
    ('Netherlands'),
    ('Switzerland');
```

## 7-Remplissage de la base de donnée

```
/* Inserting data into Supplement table */
INSERT INTO Supplement (chambre, assurance)
VALUES
('Single room', 'Travel insurance'),
('Double room', 'Cancellation insurance'),
('Triple room', 'Medical insurance'),
('Quadruple room', 'Baggage insurance'),
('Suite', 'Car rental insurance'),
('No supplement', 'No insurance'),
('Extra bed', 'Personal liability insurance'),
('Baby cot', 'Pet insurance'),
('No supplement', 'No insurance'),
('No supplement', 'No insurance');

/* Inserting data into Voyage table */
INSERT INTO Voyage (categorie, point_de_depart, dt, sejour, periode, id_typevoyage, id_pays)
VALUES
('Cruise', 'Marseille', '2023-05-01', '10 days', 'Spring', 'SEA', 2),
('Mountain', 'Chamonix', '2023-06-15', '7 days', 'Summer', 'MOT', 1),
('City break', 'Paris', '2023-07-01', '3 days', 'Summer', 'CTY', 1),
('Discovery', 'Athens', '2023-08-10', '14 days', 'Summer', 'DIS', 6),
('Sun', 'Albufeira', '2023-09-01', '10 days', 'Autumn', 'SUN', 4),
('Sea', 'Dubrovnik', '2023-10-05', '7 days', 'Autumn', 'SEA', 7),
('Cultural', 'Rome', '2024-01-15', '4 days', 'Winter', 'CUL', 3),
('Spa', 'Baden-Baden', '2024-02-15', '5 days', 'Winter', 'SPA', 5),
('Family', 'Amsterdam', '2024-04-01', '7 days', 'Spring', 'FAM', 9);

/*Inserting data into Voyage table*/
INSERT INTO Programmations (dt, id_voyage)
VALUES
('2023-05-01', 1),
('2023-06-15', 2),
('2023-07-01', 3),
('2023-08-10', 4),
('2023-09-01', 5),
('2023-10-05', 6),
('2024-01-15', 7),
('2024-02-15', 8),
('2024-04-01', 9);
```

## 7-Remplissage de la base de donnée

```
/*Inserting data into Region table*/  
INSERT INTO Region (region)  
VALUES  
    ('Tanger-Tétouan-Al Hoceïma'),  
    ('Oriental'),  
    ('Fès-Meknès'),  
    ('Rabat-Salé-Kénitra'),  
    ('Béni Mellal-Khénifra'),  
    ('Casablanca-Settat'),  
    ('Marrakech-Safi'),  
    ('Drâa-Tafilalet'),  
    ('Souss-Massa'),  
    ('Guelmim-Oued Noun'),  
    ('Laâyoune-Sakia El Hamra'),  
    ('Dakhla-Oued Ed-Dahab');  
  
/*Inserting data into Departement table*/  
INSERT INTO Departement (departement)  
VALUES  
    ('Tanger-Assilah'),  
    ('Tétouan'),  
    ('Al Hoceïma'),  
    ('Oujda-Angad'),  
    ('Nador'),  
    ('Fès'),  
    ('Meknès'),  
    ('Rabat'),  
    ('Salé'),  
    ('Kénitra'),  
    ('Beni Mellal'),  
    ('Khénifra'),  
    ('Casablanca'),  
    ('Mohammadia'),  
    ('El Jadida'),  
    ('Settat'),  
    ('Marrakech'),  
    ('Essaouira'),  
    ('Dakhla');
```

# 7-Remplissage de la base de donnée

```

/*Inserting data into Ville table*/
INSERT INTO Ville (id_voyage, nom)
VALUES
(1, 'Marseille'),
(2, 'Chamonix'),
(3, 'Paris'),
(4, 'Athens'),
(5, 'Albufeira'),
(6, 'Dubrovnik'),
(7, 'Rome'),
(8, 'Baden-Baden'),
(9, 'Amsterdam');

/*Inserting data into Client table*/
INSERT INTO Client (Nom, prenom, adresse, email, tel)
VALUES
('Benali', 'Karim', '123 Rue Mohammed V, Casablanca', 'karim.benali@gmail.com', '+212612345678'),
('El Mansouri', 'Leila', '456 Avenue Hassan II, Rabat', 'leila.elmansouri@gmail.com', '+212623456789'),
('Chaoui', 'Youssef', '789 Boulevard Mohammed VI, Marrakech', 'youssef.chaoui@gmail.com', '+212634567890'),
('Hassani', 'Lina', '321 Rue Mohammed VI, Fes', 'lina.hassani@gmail.com', '+212645678901'),
('El Amiri', 'Sara', '654 Avenue Hassan II, Tangier', 'sara.elamiri@gmail.com', '+212656789012'),
('Mahmoudi', 'Ali', '987 Boulevard Hassan II, Agadir', 'ali.mahmoudi@gmail.com', '+212667890123'),
('Bennani', 'Nour', '321 Avenue Mohammed VI, Essaouira', 'nour.bennani@gmail.com', '+212678901234'),
('El Kaddouri', 'Omar', '456 Rue Mohammed V, Meknes', 'omar.elkaddouri@gmail.com', '+212689012345'),
('Khalil', 'Hiba', '987 Boulevard Mohammed VI, Tetouan', 'hiba.khalil@gmail.com', '+212690123456'),
('El Azzouzi', 'Amir', '654 Avenue Hassan II, Oujda', 'amir.elazzouzi@gmail.com', '+212701234567');

/*Inserting data into Hotel table*/
INSERT INTO Hotel (nom_hotel)
VALUES
('Hôtel de Paris'),
('Riad Laaroussa'),
('Four Seasons Hotel Casablanca'),
('Marrakech Riad'),
('Palais Amani'),
('La Sultana Marrakech'),
('Riad Fes Maya Suite & Spa'),
('Dar Seven'),
('Hotel Sahrai'),
('Sofitel Rabat Jardin des Roses');

INSERT INTO Reservation (annulation_assu, chmbr_supp, nbr_passager, id_supp, id_prg, id_client)
VALUES
('Cancellation insurance', 'Double room', '2', 1, 1, 1),
('No insurance', 'No supplement', '1', 2, 2, 2),
('Travel insurance', 'Single room', '1', 3, 3, 3),
('Cancellation insurance', 'Extra bed', '2', 4, 4, 4),
('No insurance', 'No supplement', '1', 5, 5, 5),
('Travel insurance', 'Triple room', '3', 6, 6, 6),
('Cancellation insurance', 'Double room', '2', 7, 7, 7),
('No insurance', 'No supplement', '1', 8, 8, 8),
('Travel insurance', 'Single room', '1', 9, 9, 9);

INSERT INTO Passenger (nom, prenom, adresse, email, tel, id_reservation)
VALUES
('Zouhairi', 'Fatima', '123 Rue A', 'fatima.zouhairi@example.com', '0612345678', 21),
('El Amrani', 'Ahmed', '456 Rue B', 'ahmed.elamrani@example.com', '0698765432', 22),
('Bennani', 'Hassan', '789 Rue C', 'hassan.bennani@example.com', '0645612378', 23),
('Lamrini', 'Sanaa', '321 Rue D', 'sanaa.lamrini@example.com', '0698945612', 24),
('El Moutawakil', 'Karim', '654 Rue E', 'karim.elmoutawakil@example.com', '0612398765', 25),
('Ait Said', 'Nadia', '987 Rue F', 'nadia.aitsaid@example.com', '0698754321', 26),
('Haddad', 'Youssef', '147 Rue G', 'youssef.haddad@example.com', '0698765432', 27),
('Tazi', 'Leila', '258 Rue H', 'leila.tazi@example.com', '0612345678', 28),
('Rahmouni', 'Mounir', '369 Rue I', 'mounir.rahmouni@example.com', '0698945612', 29),
('Chami', 'Lina', '753 Rue J', 'lina.chami@example.com', '0612398765', 21);

```

## 7-Remplissage de la base de donnée

```
INSERT INTO affecter (id_voyage, id_hotel)
VALUES
  (1, 1),
  (2, 2),
  (3, 3),
  (4, 4),
  (5, 5),
  (6, 6),
  (7, 7),
  (8, 8),
  (9, 9);

INSERT INTO Monter (id_passager, id_autocar)
VALUES
  (11, 1),
  (12, 2),
  (13, 3),
  (14, 4),
  (15, 5),
  (16, 6),
  (17, 7),
  (18, 8),
  (19, 9),
  (20, 10);
```

## 8-Test de la base de donnée

Dans le cadre de notre projet de agence de voyages, nous avons effectué des tests approfondis sur notre base de données MySQL en utilisant des jointures. Les jointures nous ont permis de combiner les données provenant de différentes tables en fonction des relations établies entre elles. Grâce à ces opérations de jointure, nous avons pu obtenir des informations pertinentes et complètes en agrégeant les données de manière efficace.

Les tests effectués avec les jointures dans notre base de données ont démontré notre capacité à manipuler les données de manière précise et à extraire des informations significatives. Nous avons pu obtenir une vue d'ensemble complète de nos données en agrégeant les informations des clients, des voyages, des programmations, des passagers et des autocars.

L'utilisation des jointures a également renforcé notre capacité à réaliser des analyses approfondies et à générer des rapports détaillés sur les réservations, les participants aux voyages et les attributions d'autocars. Ces informations sont essentielles pour optimiser notre système de réservation, améliorer l'expérience des clients et prendre des décisions éclairées.



## 8-Test de la base de donnée

```
/*Jointure entre les tables Reservation et Client*/
SELECT Reservation.*, Client.Nom, Client.prenom
FROM Reservation
INNER JOIN Client ON Reservation.id_client = Client.id_client;
```

id_reservation	annulation_assu	chmbr_supp	nbr_passager	id_supp	id_prg	id_client	Nom	prenom
21	Cancellation insurance	Double room	2	1	1	1	Benali	Karim
22	No insurance	No supplement	1	2	2	2	El Mansouri	Leila
23	Travel insurance	Single room	1	3	3	3	Chaoui	Youssef
24	Cancellation insurance	Extra bed	2	4	4	4	Hassani	Lina
25	No insurance	No supplement	1	5	5	5	El Amiri	Sara
26	Travel insurance	Triple room	3	6	6	6	Mahmoudi	Ali
27	Cancellation insurance	Double room	2	7	7	7	Bennani	Nour
28	No insurance	No supplement	1	8	8	8	El Kaddouri	Omar
29	Travel insurance	Single room	1	9	9	9	Khalil	Hiba

```
/*Jointure entre les tables Voyage, Programmmations et Pays*/
SELECT Voyage.*, Programmmations.dt, Pays.pays
FROM Voyage
INNER JOIN Programmmations ON Voyage.id_voyage = Programmmations.id_voyage
INNER JOIN Pays ON Voyage.id_pays = Pays.id_pays;
```

id_voyage	categorie	point_de_depart	dt	sejour	periode	id_typevoyage	id_pays	dt	pays
1	Cruise	Marseille	2023-05-01	10 days	Spring	SEA	2	2023-05-01	Spain
2	Mountain	Chamonix	2023-06-15	7 days	Summer	MOT	1	2023-06-15	France
3	City break	Paris	2023-07-01	3 days	Summer	CTY	1	2023-07-01	France
4	Discovery	Athens	2023-08-10	14 days	Summer	DIS	6	2023-08-10	Greece
5	Sun	Albufeira	2023-09-01	10 days	Autumn	SUN	4	2023-09-01	Portugal
6	Sea	Dubrovnik	2023-10-05	7 days	Autumn	SEA	7	2023-10-05	Croatia
7	Cultural	Rome	2024-01-15	4 days	Winter	CUL	3	2024-01-15	Italy
8	Spa	Baden-Baden	2024-02-15	5 days	Winter	SPA	5	2024-02-15	Germany
9	Family	Amsterdam	2024-04-01	7 days	Spring	FAM	9	2024-04-01	Netherlands

En conclusion, les tests effectués sur notre base de données MySQL en utilisant des jointures ont été fructueux. Ils ont démontré notre maîtrise des concepts de jointure et notre aptitude à exploiter pleinement les capacités de notre base de données. Ces résultats nous permettront d'optimiser notre système de réservation de voyages et de fournir des services de haute qualité à nos clients.

## 9-Conclusion

Dans le cadre de ce projet, j'ai assumé un rôle autonome et individuel, prenant en charge l'intégralité des tâches et responsabilités selon les directives établies dans le cahier des charges. En qualité de seul contributeur, j'ai pu mettre en pratique les connaissances et compétences que j'ai acquises tout au long de ma formation. Cette expérience a été l'occasion de développer ma capacité à travailler de manière indépendante, à prendre des décisions et à résoudre des problèmes de manière autonome. J'ai appliqué mes compétences spécifiques en matière de bases de données ainsi que dans d'autres domaines pertinents pour ce projet. Travailler seul m'a également permis de gérer efficacement mon temps et de prendre des décisions en fonction de mes propres capacités et contraintes.