**Conception et Réalisation d'une base de Données de Location de Véhicule d'une Agence**

## Introduction

Ce rapport présente la conception, la structure et l'exploitation d'une base de données pour une agence de location de véhicules. Le système permet de gérer les clients, les véhicules, les contrats, les agences, les concessions, et les employés. L'objectif est d'assurer un suivi efficace des ventes, des locations, et de la maintenance des véhicules.

## Objectif du Projet

* Concevoir une base de données respectant la 3e forme normale (3FN).
* Gérer les relations entre clients, employés, contrats, véhicules et concessions.
* Fournir des requêtes SQL d'analyse pour l'exploitation commerciale et technique.

## Prompt Utilisé :

Conçois une base de données complète pour une agence spécialisée dans la location de véhicules.  
L’objectif principal du projet est de permettre la gestion centralisée et automatisée de toutes les opérations liées à la location de véhicules, en assurant la traçabilité des clients, des véhicules, des contrats de location, des paiements, des employés, et des catégories de véhicules.  
Le système doit répondre aux besoins opérationnels, commerciaux et administratifs de l’agence, tout en garantissant la cohérence, la sécurité et la fiabilité des données.

Fournis :

1. Recueillir et formaliser les règles métier qui structurent le fonctionnement de l’agence, en tenant compte des processus de location, de gestion des clients, de suivi des véhicules, de gestion des employés, de facturation et de paiement.
2. Établir un dictionnaire de données exhaustif, listant toutes les tables nécessaires, leurs attributs, les types de données associés, les tailles, ainsi que les clés primaires et étrangères. Ce dictionnaire doit permettre de comprendre la nature et le rôle de chaque donnée.
3. Et un modèle conceptuel de données (MCD**)** conforme à la 3FN.

**Les contraintes à respecter :**

* Un client peut louer plusieurs véhicules, mais un contrat de location ne concerne qu’un seul véhicule.
* Un véhicule peut être loué plusieurs fois à des dates différentes.
* Chaque location est gérée par un employé.
* Chaque véhicule appartient à une catégorie (citadine, SUV, utilitaire, etc.).
* La facture est émise pour chaque contrat de location.

Fournis ta réponse sous forme claire et structurée.

## Réponse de l’IAG :

#### Règles métier

* Un **client** est identifié par un numéro unique et peut effectuer plusieurs **locations**.
* Une **location** concerne un seul **véhicule**, mais un véhicule peut être loué plusieurs fois.
* Une **location** est gérée par un seul **employé** de l’agence.
* Chaque **véhicule** appartient à une **catégorie** (ex. : citadine, berline, SUV…).
* Une **facture** est émise pour chaque **contrat de location**.
* Un **paiement** est associé à une seule **facture**.
* Les **dates de début et de fin de location** doivent être cohérentes (date\_fin > date\_début).
* Un **véhicule** ne peut être réservé que s’il est **disponible**.

#### Dictionnaire de données

**Ce dictionnaire de données décrit la structure de la base de données de l’agence de location de véhicules. Il présente les différentes tables, leurs attributs, les types de données associés, ainsi que les clés primaires et étrangères.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table** | **Attribut** | **Type** | **Clé** | **Description** |
| **Client** | **id\_client** | **INT** | **PK** | **Identifiant unique du client** |
| **Client** | **nom** | **VARCHAR(50)** |  | **Nom du client** |
| **Client** | **prenom** | **VARCHAR(50)** |  | **Prénom du client** |
| **Client** | **adresse** | **VARCHAR(100)** |  | **Adresse postale** |
| **Client** | **telephone** | **VARCHAR(15)** |  | **Numéro de téléphone** |
| **Employe** | **id\_employe** | **INT** | **PK** | **Identifiant de l’employé** |
| **Employe** | **nom** | **VARCHAR(50)** |  | **Nom de l’employé** |
| **Employe** | **poste** | **VARCHAR(30)** |  | **Fonction dans l’agence** |
| **Categorie** | **id\_categorie** | **INT** | **PK** | **Identifiant de catégorie** |
| **Categorie** | **libelle** | **VARCHAR(30)** |  | **Nom de la catégorie** |
| **Vehicule** | **id\_vehicule** | **INT** | **PK** | **Identifiant unique du véhicule** |
| **Vehicule** | **marque** | **VARCHAR(50)** |  | **Marque du véhicule** |
| **Vehicule** | **modele** | **VARCHAR(50)** |  | **Modèle du véhicule** |
| **Vehicule** | **immatriculation** | **VARCHAR(20)** |  | **Numéro d’immatriculation** |
| **Vehicule** | **id\_categorie** | **INT** | **FK** | **Catégorie du véhicule** |
| **Vehicule** | **disponible** | **BOOLEAN** |  | **Indique si le véhicule est disponible** |
| **Location** | **id\_location** | **INT** | **PK** | **Identifiant unique de la location** |
| **Location** | **id\_client** | **INT** | **FK** | **Client concerné** |
| **Location** | **id\_vehicule** | **INT** | **FK** | **Véhicule loué** |
| **Location** | **id\_employe** | **INT** | **FK** | **Employé responsable** |
| **Location** | **date\_debut** | **DATE** |  | **Début de la location** |
| **Location** | **date\_fin** | **DATE** |  | **Fin de la location** |

## MCD Simplifié :

* Client (id\_client, nom, prenom, adresse, telephone)
* Employe (id\_employe, nom, poste)
* Categorie (id\_categorie, libelle)
* Vehicule (id\_vehicule, marque, modele, immatriculation, disponible, id\_categorie)
* Location (id\_location, date\_debut, date\_fin, montant\_total, id\_client, id\_vehicule, id\_employe)
* Facture (id\_facture, date\_facture, total, id\_location)
* Paiement (id\_paiement, mode\_paiement, date\_paiement, id\_facture)

## Prompt pour l’insertion

Tu es un assistant spécialisé en génération de données réalistes pour une base de données SQL.  
Je souhaite que tu génères automatiquement des instructions INSERT INTO pour remplir les tables suivantes :

* **agence(a\_nom, a\_pays, a\_ville)** : 30 enregistrements, agences françaises dans différentes grandes villes.
* **conces(c\_nom, c\_pays, c\_type, c\_ville)** : 30 enregistrements, vrais concessionnaires automobiles (marques internationales).
* **maison(m\_marque, m\_date)** : 30 marques automobiles réelles avec des dates de création comprises entre 2016 et 2020.
* **adresse(ad\_adresse, ad\_num\_rue, ad\_nom\_rue, ad\_codepostal)** : 30 adresses françaises réalistes.
* **emploiye(e\_id, e\_nom, e\_prenom, e\_mail, e\_age, ad\_adresse, a\_nom)** :30 employés avec noms internationaux, âges entre 25 et 50 ans, e-mails professionnels, et lien logique avec les agences et adresses.
* **clients(cl\_id, cl\_nom, cl\_prenom, cl\_date, cl\_age, cl\_email, ad\_adresse, e\_id, permis)** : 30 clients différents, âge 25–50, années d’inscription > 2015, “permis = ‘oui’”.
* **contrat(co\_id, co\_type, co\_date\_debut, co\_date\_fin, co\_statut\_paiement, co\_etat\_départ, co\_etat\_fin, cl\_id)** : 30 contrats cohérents avec les clients, types variés (“achat”, “location simple”, “location avec option d’achat”).
* **voiture(v\_id, v\_etat, v\_couleur, v\_prix, v\_marque, v\_cheveaux, v\_type, v\_autonomie, co\_id, c\_nom)** : 30 voitures correspondant à de vrais modèles et marques, avec prix et caractéristiques réalistes.
* **utilise(a\_nom, c\_nom)** : 30 associations logiques entre agences et concessionnaires.
* **Contraintes de génération :**
* Toutes les valeurs doivent être **cohérentes et réalistes** (pas de doublons, emails corrects, marques existantes).
* Le format final doit être **directement exécutable dans SQL**, sous forme de blocs INSERT INTO ... VALUES (...).
* Le texte généré doit être enregistré dans un fichier nommé **3\_insertion.sql**.
* Respecter l’ordre logique des dépendances : d’abord les tables sans clés étrangères, ensuite celles avec FKs
* **Style attendu :**
* Commentaires clairs séparant les sections (-- ==========================).
* Données variées, réalistes et équilibrées (nationalités, marques, villes, âges, prix, etc.).

## Requête d’interrogation

Scénario 1 : Analyse commerciale

1. Trouver les 10 clients les plus récents ayant fourni un permis de conduire valide.
2. Calculer le pourcentage de chaque type de contrat (achat, location, leasing) sur le total des ventes.
3. Identifier la marque de voiture ayant le plus de contrats signés.
4. Compter le nombre de contrats signés par chaque agence en 2021.
5. Lister les clients âgés de 25 à 35 ans enregistrés depuis 2018.
6. Afficher les voitures sans contrat actif avec leurs prix et caractéristiques.
7. Calculer la moyenne du prix et de la puissance pour chaque type de véhicule.
8. Trouver les concessions travaillant avec plusieurs marques différentes.
9. Identifier l’agence ayant signé le plus grand nombre de contrats.
10. Suivre l’évolution du nombre de contrats par année depuis 2016.

Scénario 2 : Suivi opérationnel et maintenance

1. Trouver les véhicules n’ayant jamais eu de contrat.
2. Identifier les 5 véhicules les plus utilisés dans chaque catégorie.
3. Repérer les concessions possédant des véhicules stockés depuis plus d’un an sans contrat.
4. Lister les modèles présents dans plusieurs concessions.
5. Trouver les clients ayant loué tous les modèles de type SUV.
6. Afficher la date de la dernière maintenance de chaque véhicule ou NULL si aucune n’a été faite.
7. Trouver les véhicules dont le prix est supérieur à celui de toutes les citadines.
8. Identifier les agences sans aucune opération d’entretien depuis 2023.
9. Lister les véhicules moins chers que certains SUV.
10. Trouver les clients sans contrat actif depuis plus d’un an.

## Conclusion

Ce projet a permis de développer une base de données robuste et normalisée pour une agence de location de véhicules. Les requêtes SQL réalisées offrent un double objectif : aider à la décision commerciale et optimiser la gestion technique du parc. Ce travail illustre la capacité de la modélisation et de l'interrogation SQL à répondre à des besoins réels d'entreprise.