Université de Sousse

École Nationale d'Ingénieurs de Sousse

Rapport du projet module Base de données

Aéroport

Elaboré par :

Hela BEN HNIA

Nouha ESSID

Enseignant :

Mr Mohamed Nazih Omri

2éme année Informatique appliquée 1
2022/2023

Sommaire

Introduction Générale

Une base de données est un ensemble organisé d'informations avec un objectif commun.

Peu importe le support utilisé pour rassembler et stocker les données (papier, fichiers, etc.), dès que des données sont rassemblées et stockées d'une manière organisée dans un but spécifique, on parle de base de données qui sera ainsi accessible à la demande pour plusieurs utilisateurs et des besoins divers.

Chapitre 1

Modèle entité-association

1.1 Introduction

Dans ce chapitre on va présenter le modèle entité-association de notre projet intitulé « Aéroport ».

1.2 Modèle

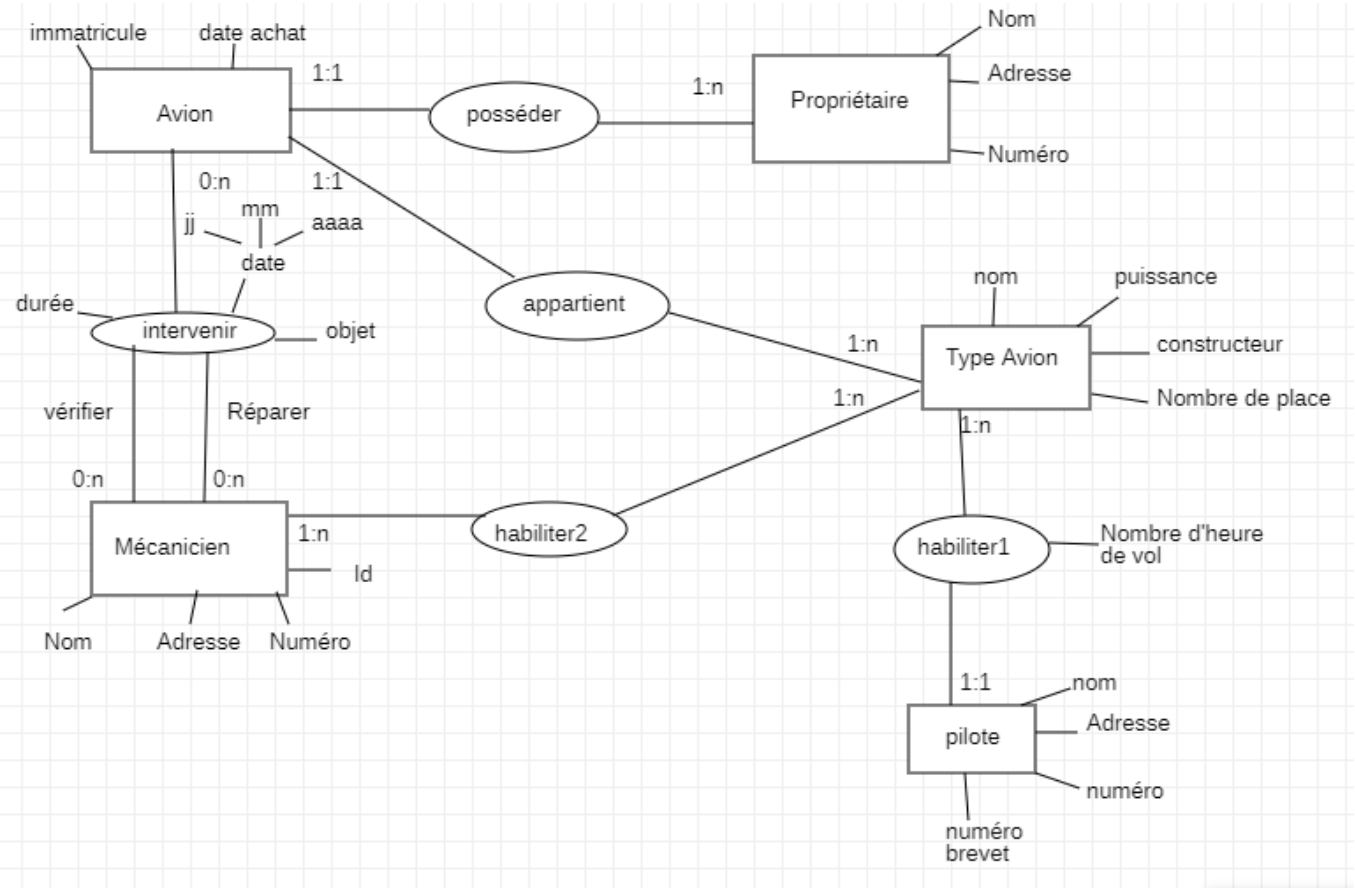


Figure1 : Modèle entité-association

- ✓ Posséder, intervenir, appartient, habiliter1 et habiliter2 sont des types d'association.
- ✓ Avion, pilote, mécanicien, type avion et propriétaire **des types d'entité.**

1.3 Conclusion

Dans ce chapitre, on a étudié les détails du projet et les présentés sous forme d'un modèle entité-association, pour mieux comprendre les principales tâches.

Chapitre2

Modèle Relationnel

2.1 Introduction

Le chapitre intitulé Modèle Relationnel a pour but de présenter le modèle entité- association précèdent sous forme des relations entre les différentes entités.

2.2 Modèle relationnel

Pour avoir une transformation d'un modèle Entité-Association M.E/A en un modèle Relationnel MR il faut respecter 3 règles de transformation :

Règle 1 : Tout type d'Entité TE se transforme en une relation.

Règle 2 : En présence de cardinalité du type m : n, m : n le type Association TA en question se transforme en une relation R.

Règle 3 : En présence de cardinalité du type 0 :1 ou 1 :1, le type Association TA ne se transforme pas en une relation mais on ajoute tous les identifiants de typeEntité coté cardinalité m : n au type Entité coté cardinalité type 0 :1 ou 1 :1.

Après l'application de ces 3 règles on trouve les relations suivantes :

Propriétaire(IdProp, nom, adresse, numéro)

Avion (Immatricule, date-achat, #IdProp, #nomT)

#IdProp référence à l'identifiant du propriétaire.

#nomT référence au type.

Mécanicien (idM, nom, num-téléphone, adresse)

idM référence à l'identifiant du mécanicien.

Intervention(#immatricule, #IdR, #IdV, objet, date, durée)

#Immatricule référence Avion

#IdV réfère au mécanicien de vérification.

#IdR réfère au mécanicien de réparation.

Type (nomT, constructeur, puissance, nombre-place)

nomT référence le type d'avion.

Pilote (IdP, nom, adresse, numBrev, numTel)

IdP référence le pilote.

Habilitation1(#nomT, #IdP, nbr-heure-vol)

#IdP référence le pilote.

#nomT référence le type d'avion.

Habilitation2 (#IdM, #nomT)

#IdM référence le mécanicien.

#nomT référence le type d'avion.

2.3 Conclusion

Dans ce chapitre on a effectué la présentation du modèle relationnel de notre projet « Aéroport » dans lequel on a parlé des trois règles qui nous ramènent à effectuer une transformation du modèle Entité-Association en un modèle relationnel.

Chapitre3

Normalisation

3.1 Introduction

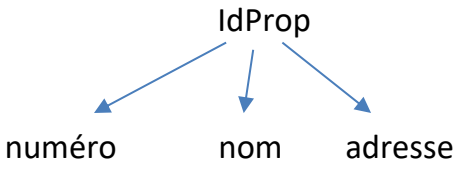
Dans ce chapitre on va normaliser notre base de données de notre projet.

3.2 Les relations

➢ Normalisation

Propriétaire(*IdProp*, adresse, nom, numéro)

Graphe minimum :



Redondances :

- Pas de redondances

Identifiant de la relation :

- Id =(IdProp)

Forme normale :

1FN :

- La relation est en 1FN car tous les attributs de cette relation sont simples et monovalués en supposant que l'adresse est une chaine de caractères et le numéro de téléphone est monovalué.

2FN :

- La relation est en 2FN car elle est en 1FN et chaque attribut qui ne fait pas partie de l'identifiant dépend de l'identifiant entier(IdProp).

3FN :

- Chaque attribut qui ne fait pas partie d'identifiant dépend directement de l'identifiant entier(IdProp).

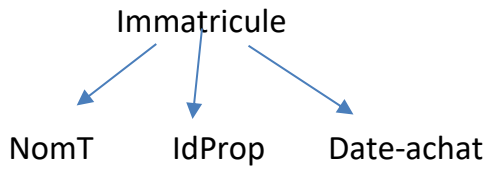
FNBC :

- La source de DF (IdProp) est un identifiant entier.

4FN :

- La relation est en 4FN car il n'y a pas de dépendances fonctionnels multivalués.

Avion (Immatricule, date-achat, #IdProp, #nomT)
Graphe minimum :



Redondances :

- Pas de redondances

Identifiant de la relation :

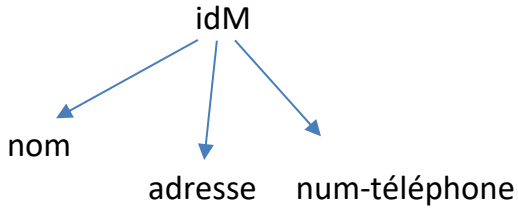
- Id= (Immatricule)

Forme normale

- **1FN :** La relation est en 1FN car tous les attributs de cette relation sont simples et monovalués .
- **2FN :** La relation est en 2FN car elle est en 1FN et chaque attribut qui ne fait pas partie de l'identifiant dépend de l'identifiant entier(Immatricule).
- **3FN :** Chaque attribut qui ne fait pas partie d'identifiant dépend directement de l'identifiant entier(Immatricule).
- **FNBC :** La source de DF (Immatricule) est un identifiant entier.
- **4FN :** La relation est en 4FN car il n'y a pas de dépendances fonctionnels multivalués.

Mécanicien (idM, nom, num-téléphone, adresse)

Graphe minimum :



Redondances :

- Pas de redondances.

Identifiant de la relation :

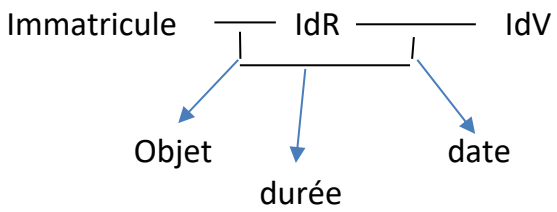
- id =(idM)

Forme normale

- **1FN :** La relation est en 1FN car tous les attributs de cette relation sont simples et monovalués en supposant que l'adresse est une chaine de caractères et le numéro de téléphone est monovalué.
- **2FN :** La relation est en 2FN car elle est en 1FN et chaque attribut qui ne fait pas partie de l'identifiant dépend de l'identifiant entier(IdM).
- **3FN :** Chaque attribut qui ne fait pas partie d'identifiant dépend directement de l'identifiant entier(IdM).
- **FNBC :** La source de DF (IdM) est un identifiant entier.
- **4FN :** La relation est en 4FN car il n'y a pas de dépendances fonctionnels multivalués.

Intervention(#immatricule, #IdR, #IdV, objet, date, durée)

Graphe minimum :



Redondances :

- Pas de redondances

Identifiant de la relation :

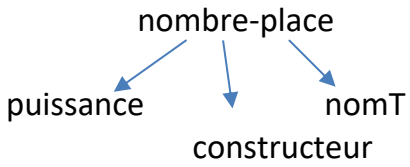
- Id=(Immatricule,#IdR,#IdV)

Forme normale :

- **1FN :** La relation est en 1FN car tous les attributs de cette relation sont simples et monovalués.
- **2FN :** La relation est en 2FN car elle est en 1FN et chaque attribut qui ne fait pas partie de l'identifiant dépend de l'identifiant entier.
- **3FN :** Chaque attribut qui ne fait pas partie d'identifiant dépend directement de l'identifiant entier.
- **FNBC :** La source de DF est un identifiant entier.
- **4FN :** La relation est en 4FN car il n'y a pas de dépendances fonctionnels multivalués.

Type (nomT, constructeur, puissance, nombre-place)

Graphe minimum



Redondances :

- Pas de redondances

Identifiant de la relation :

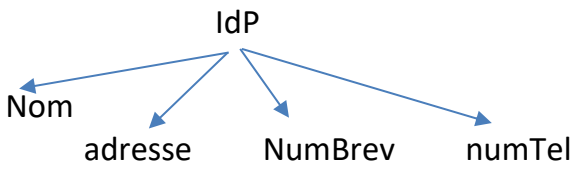
- Id=(nomT)

Forme normale :

- **1FN :** La relation est en 1FN car tous les attributs de cette relation sont simples et monovalués.
- **2FN :** La relation est en 2FN car elle est en 1FN et chaque attribut qui ne fait pas partie de l'identifiant dépend de l'identifiant entier.
- **3FN :** Chaque attribut qui ne fait pas partie d'identifiant dépend directement de l'identifiant entier.
- **FNBC :** La source de DF est un identifiant entier.
- **4FN :** La relation est en 4FN car il n'y a pas de dépendances fonctionnels multivalués.

Pilote (IdP, nom, adresse, numBrev, numTel)

Graphe minimum :



Redondances :

- Pas de redondances

Identifiant de la relation :

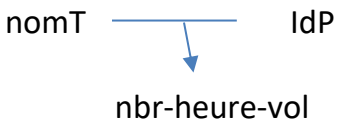
- Id=(IdP)

Forme normale :

- **1FN :**La relation est en 1FN car tous les attributs de cette relation sont simples et monovalués.
- **2FN :** La relation est en 2FN car elle est en 1FN et chaque attribut qui ne fait pas partie de l'identifiant dépend de l'identifiant entier.

- **3FN :** Chaque attribut qui ne fait pas partie d'identifiant dépend directement de l'identifiant entier.
- **FNBC :** La source de DF est un identifiant entier.
- **4FN :** La relation est en 4FN car il n'y a pas de dépendances fonctionnels multivalués.

Habilitation1(<u>#nomT, #IdP</u>, nbr-heure-vol)
Graphe minimum :



Redondances :

- Pas de redondances

Identifiant de la relation :

- Id=(<u>#nomT, #IdP</u>)

Forme normale :

- **1FN :** La relation est en 1FN car tous les attributs de cette relation sont simples et monovalués.
- **2FN :** La relation est en 2FN car elle est en 1FN et chaque attribut qui ne fait pas partie de l'identifiant dépend de l'identifiant entier.
- **3FN :** Chaque attribut qui ne fait pas partie d'identifiant dépend directement de l'identifiant entier.
- **FNBC :** La source de DF est un identifiant entier.
- **4FN :** La relation est en 4FN car il n'y a pas de dépendances fonctionnels multivalués.

Habilitation2 (<u>#IdM, #nomT</u>)
Graphe minimum :

Redondances :

- Pas de redondances.

Identifiant de la relation :

- id =(<u>#IdM, #nomT</u>)

Forme normale

- **1FN :** La relation est en 1FN car tous les attributs de cette relation sont simples et monovalués en supposant que l'adresse est une chaine de caractères et le numéro de téléphone est monovalué.
- **2FN :** La relation est en 2FN car elle est en 1FN et chaque attribut qui ne fait pas partie de l'identifiant dépend de l'identifiant entier.

- **3FN :** Chaque attribut qui ne fait pas partie d'identifiant dépend directement de l'identifiant entier.
- **FNBC :** La source de DF est un identifiant entier.
- **4FN :** La relation est en 4FN car il n'y a pas de dépendances fonctionnels multivalués.

3.3 Conclusion :

Dans ce chapitre, on a effectué la normalisation de notre projet « aéroport » afin d'éviter la redondance et d'assurer l'intégrité des données.
Ceci facilite également la maintenance de la base de données.

Chapitre 4

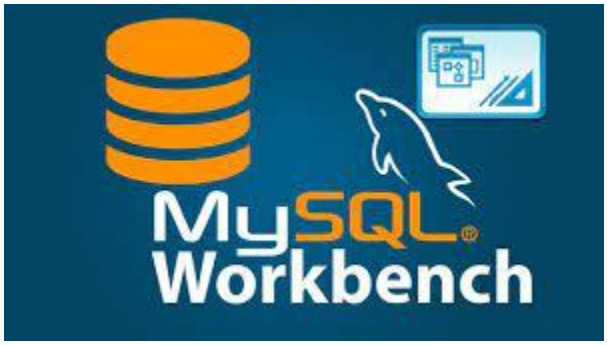
Implémentation

4.1 Introduction

Dans ce 4éme chapitre on va faire l'implémentation de notre projet en SQL et présenter l'outil qu'on a utilisé pendant cette étape.

4.2 Outils utilisés

Au cours de cette implémentation on a fait recours à MySQL.
MySQL Workbench est un logiciel de gestion et d'administration de bases de données MySQL. Via une interface graphique intuitive, il permet, entre autres, de créer, modifier ou supprimer des tables, des comptes utilisateurs, et d'effectuer toutes les opérations inhérentes à la gestion d'une base de données. Pour ce faire, il doit être connecté à un serveur MySQL.



Le logo de MySQL Workbench

4.3 Implémentation de l'application

Dans cette figure on présente les tables de notre base de données :

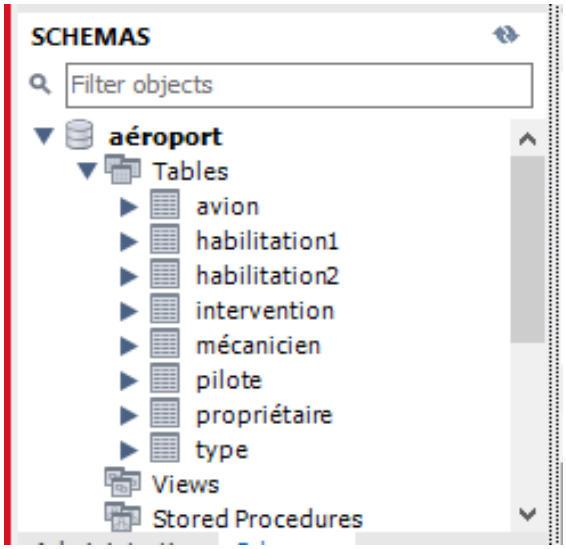


Figure des tables de la base de données

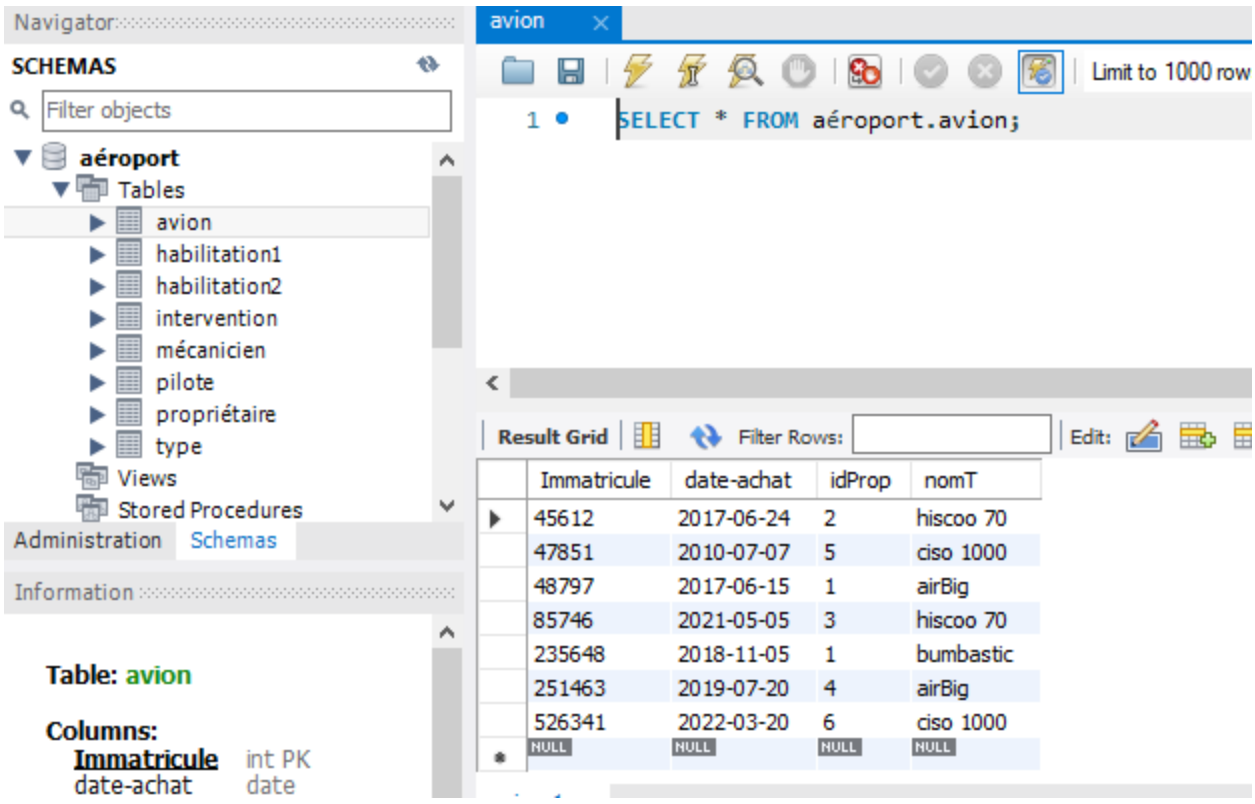


Figure : Le contenu de la table avion

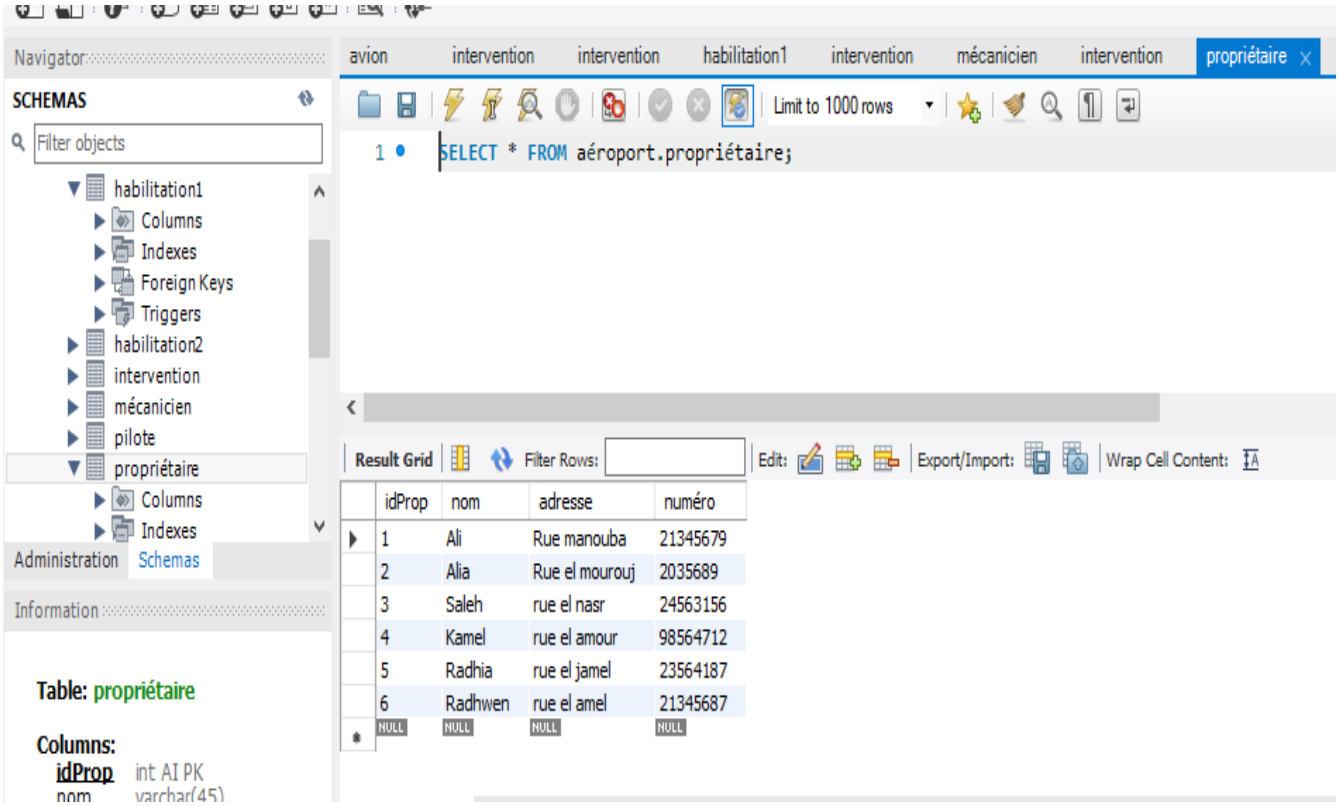
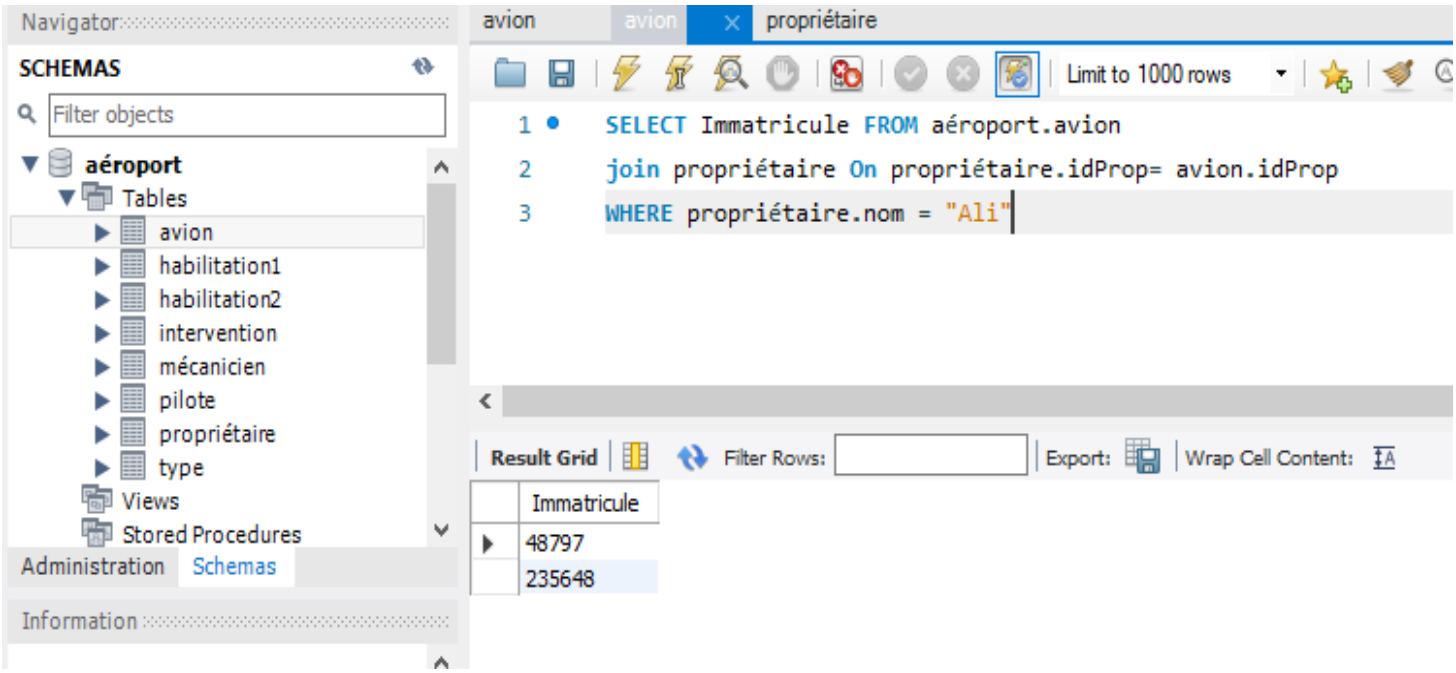


Figure : Le contenu de la table propriétaire

- Requête : Afficher les immatricules des avions dont le nom de propriétaire est « Ali » :



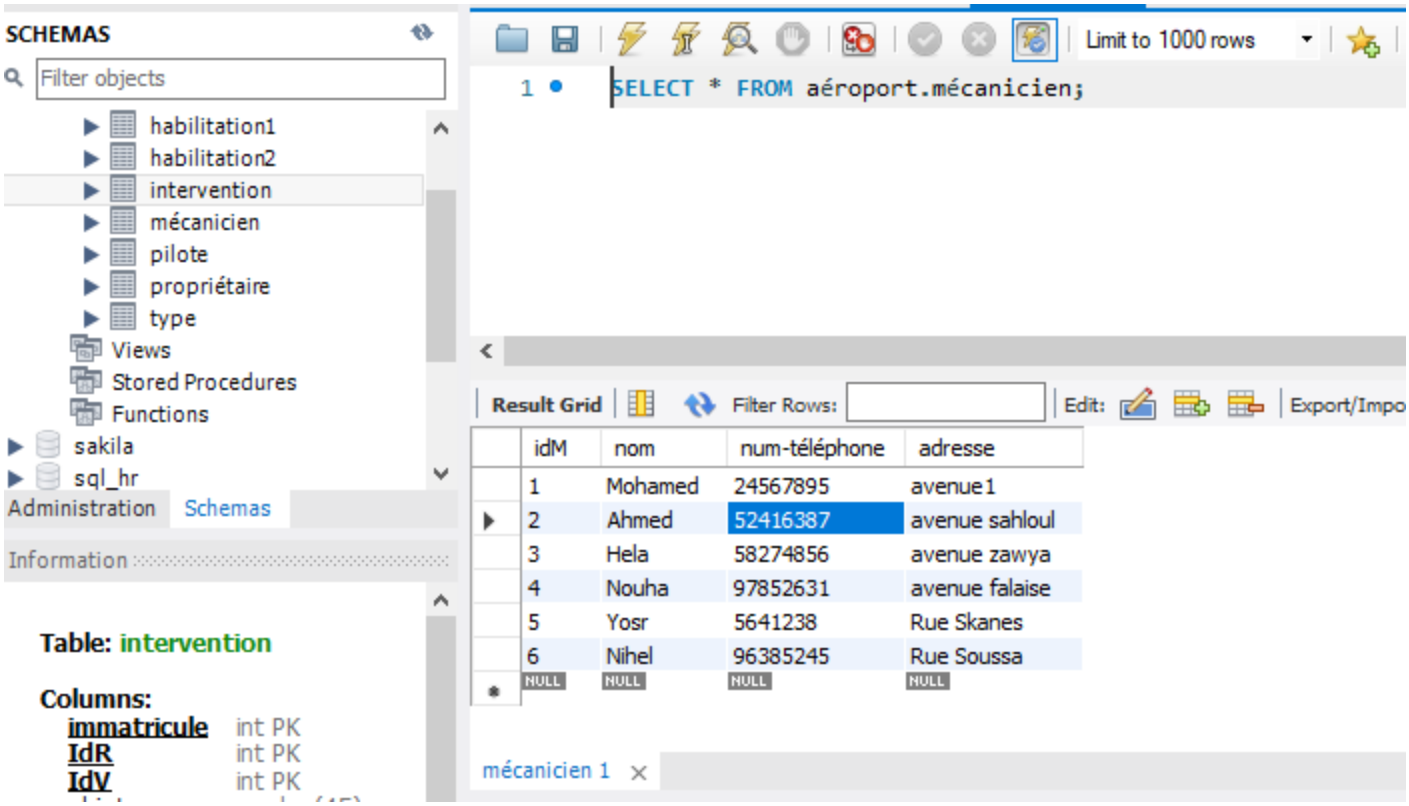


Figure : Le contenu de la table mécanicien

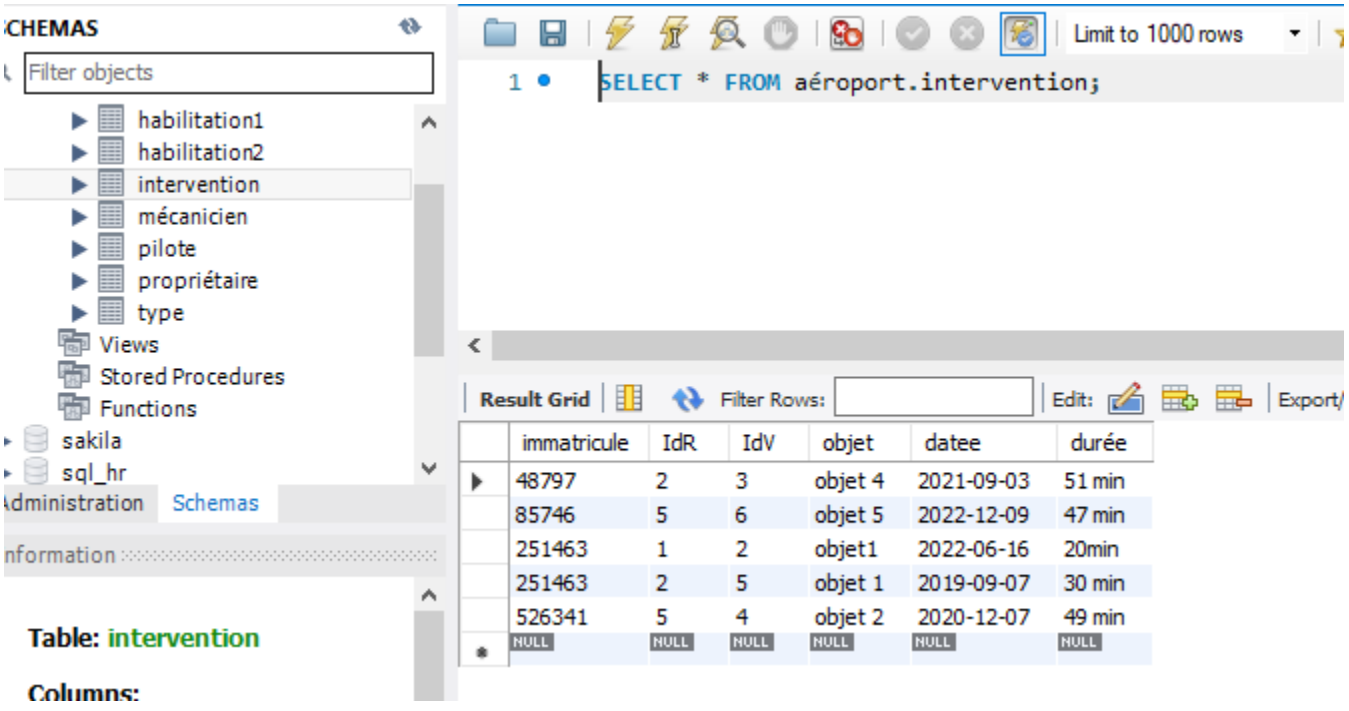
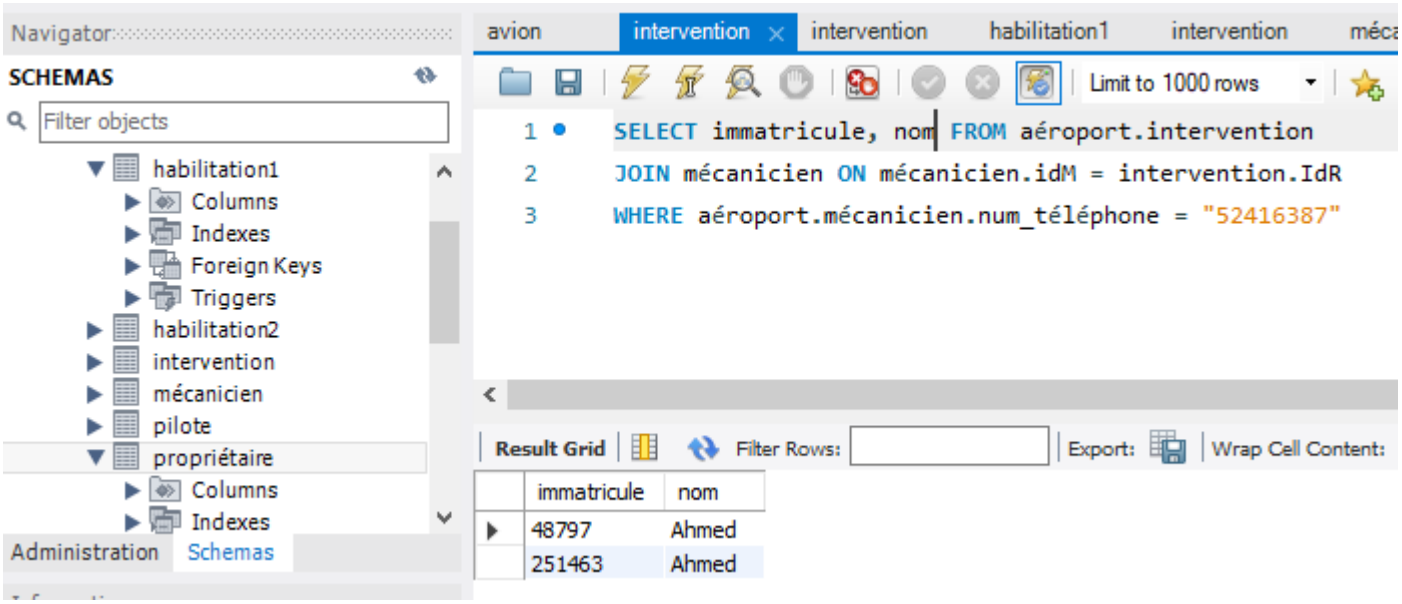


Figure : Le contenu de la table intervention

- Requête : Afficher l'immatricule et le nom du mécanicien dont le numéro de téléphone = « 52416387 »



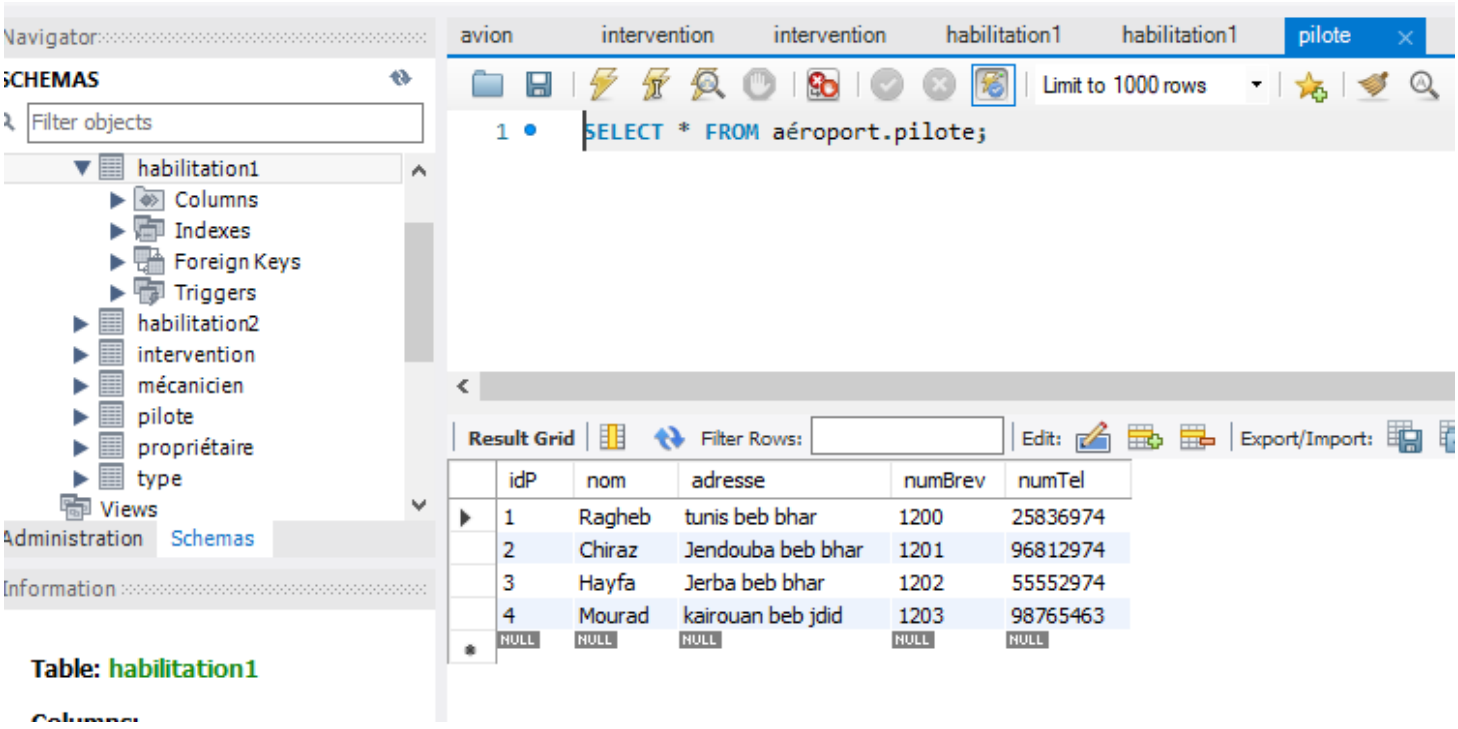


Figure : Le contenu de la table pilote

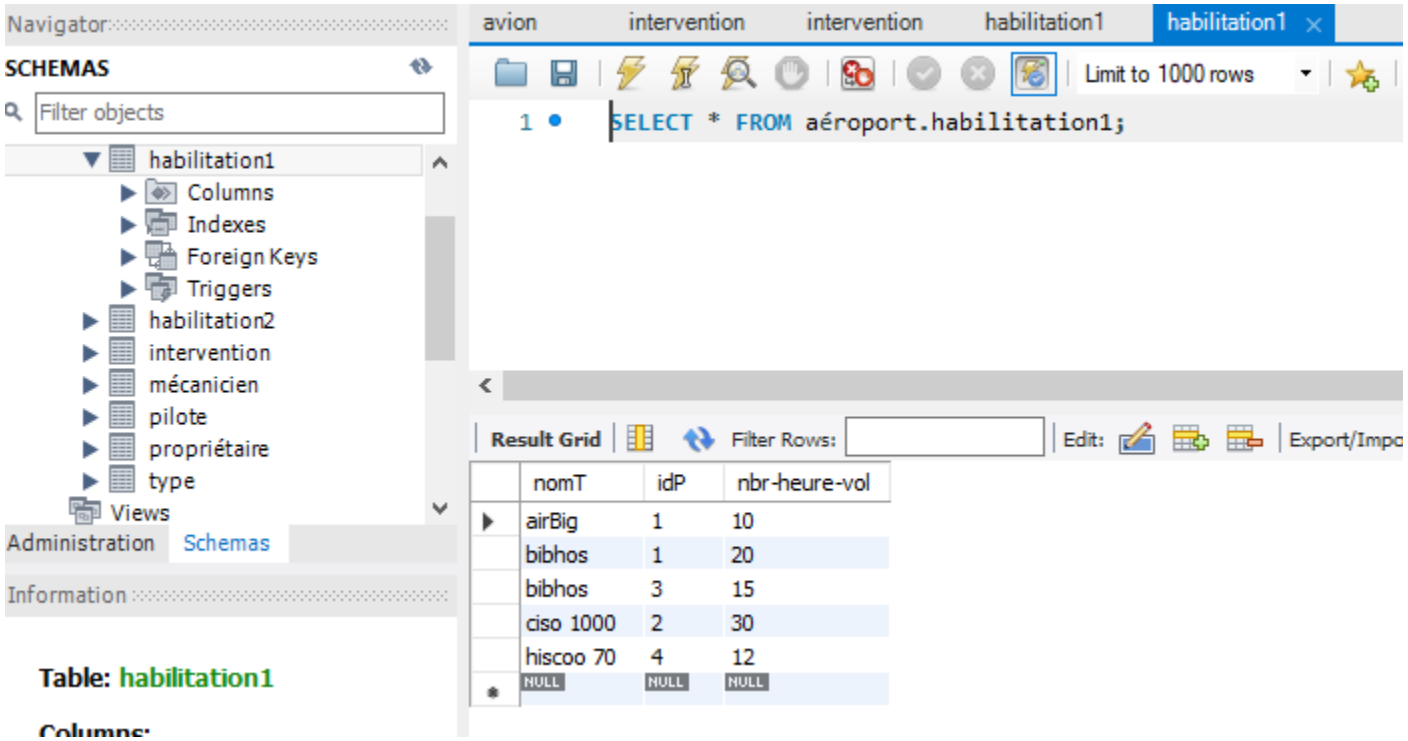
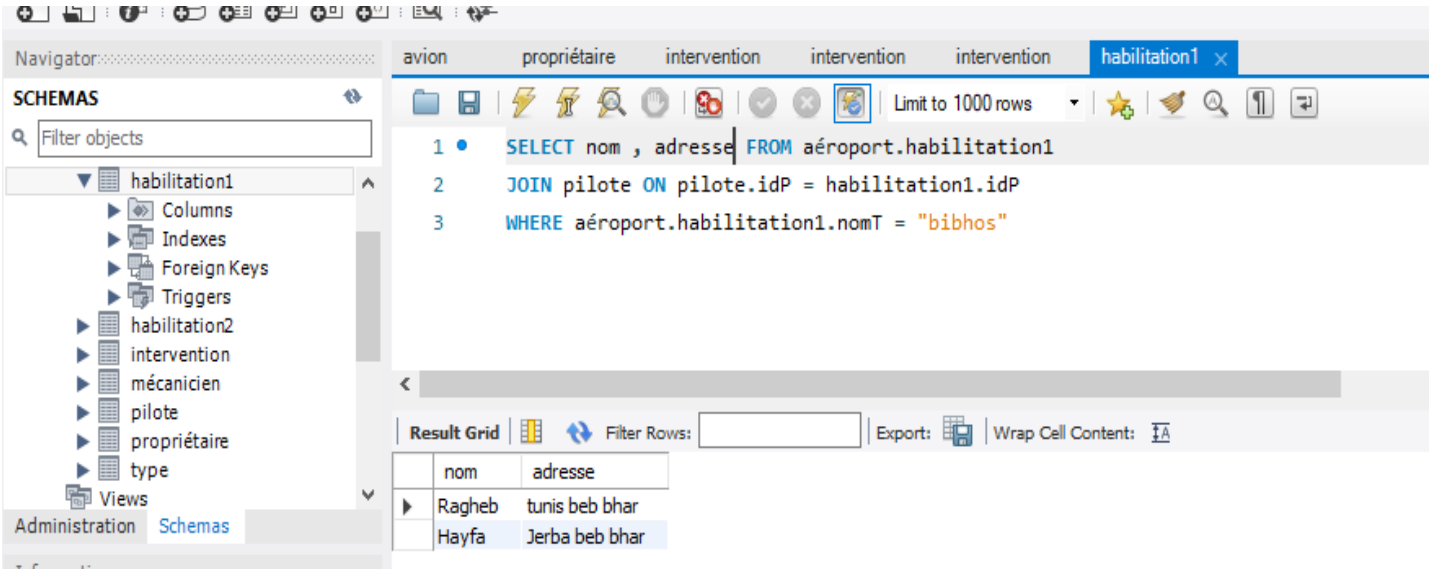


Figure : Le contenu de la table habilitation

- Requête : Afficher le nom et l'adresse des pilotes qui sont habilités sur l'avion de type « bibhos»



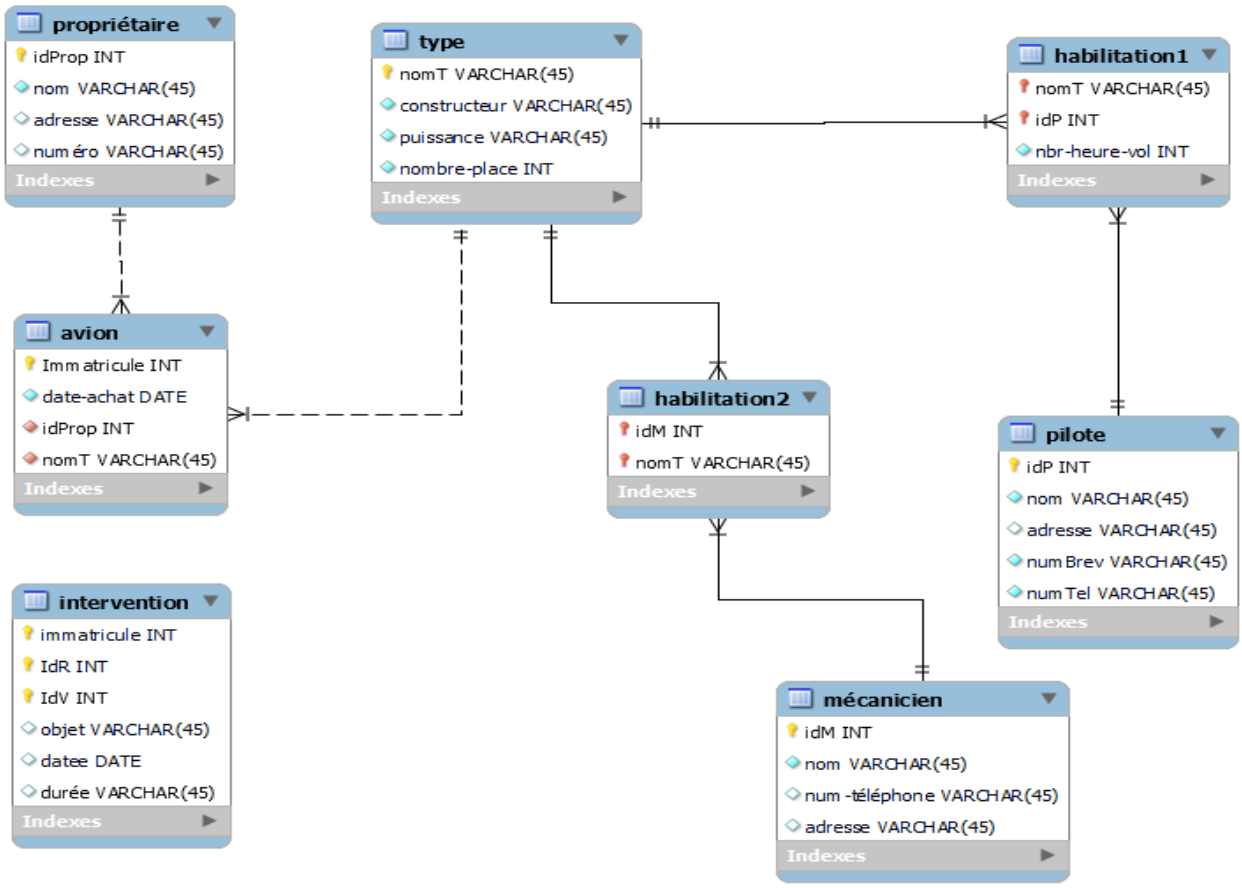


Figure : Vue Relationnelle de la base

4.4 Conclusion

Dans ce chapitre on a réalisé l'implémentation de la base de données de problème étudié dans les chapitres précédents en utilisant MySQL Workbench comme système de gestion de bases de données relationnelles.

Conclusion Générale

Grâce à ce projet on a acquis de nombreuses compétences précieuses tel que l'implémentation de la base de données sur le serveur local et le SQL qui nous seront très utiles dans nos futurs projets.

On souhaite ainsi dans une seconde étape développer un site web ou application mobile et la connectée avec notre base de données.