

- ❖ Après avoir construit le diagramme EER du projet 'Jarvis', nous tenons à créer toute la table qui s'y trouvent dans MLD vers MPD. Mais avant toute chose on doit créer la base des données 'Jarvis'.

CREATION DE LA BASE DES DONNEES

```
CREATE DATABASE IF NOT EXISTS `Jarvis` ;
```

CREATION DE TABLE

Table Proprietaire

```
CREATE TABLE IF NOT EXISTS `Jarvis`.`Proprietaire` (  
  `Utilisateur_id_U` INT NOT NULL,  
  `id_proprietaire` VARCHAR(45) NOT NULL,  
  `lien_parente` VARCHAR(45) NULL,  
  PRIMARY KEY (`Utilisateur_id_U`, `id_proprietaire`),  
  CONSTRAINT `fk_Proprietaire_Utilisateur1`  
    FOREIGN KEY (`Utilisateur_id_U`)  
    REFERENCES `Jarvis`.`Utilisateur` (`id_U`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB
```

Tables Utilisateur

```
CREATE TABLE IF NOT EXISTS `Jarvis`.`utilisateur` (  
  `id_U` INT NOT NULL,  
  `nom_U` VARCHAR(45) NOT NULL,  
  `date_naissance_U` DATE NULL,  
  `sexe_U` VARCHAR(1) NOT NULL,  
  `mot_de_passe_U` VARCHAR(10) NOT NULL,  
  `email_U` VARCHAR(30) NOT NULL,  
  PRIMARY KEY (`id_U`),  
  REFERENCES `Jarvis`.`Utilisateur` (`id_U`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB
```

Table Co-propietaire

```
CREATE TABLE IF NOT EXISTS `Jarvis`.`Co_proprietaire` (  
  `Utilisateur_id_U` INT NOT NULL,  
  `id_Co_proprietaire` INT NOT NULL,  
  `lien_parente` VARCHAR(45) NULL,  
  PRIMARY KEY (`Utilisateur_id_U`, `id_Co_proprietaire`),  
  CONSTRAINT `fk_Co-propietaire_Utilisateur1`  
    FOREIGN KEY (`Utilisateur_id_U`)  
    REFERENCES `Jarvis`.`Utilisateur` (`id_U`)  
    ON DELETE NO ACTION
```

ON UPDATE NO ACTION)

ENGINE = InnoDB

Table Membres

```
CREATE TABLE IF NOT EXISTS `Jarvis`.`Membres` (  
  `Utilisateur_id_U` INT NOT NULL,  
  `id_membres` INT NOT NULL,  
  `lien_parente` VARCHAR(45) NULL,  
  PRIMARY KEY (`Utilisateur_id_U`, `id_membres`),  
  CONSTRAINT `fk_Membres_Utilisateur`  
    FOREIGN KEY (`Utilisateur_id_U`)  
    REFERENCES `Jarvis`.`Utilisateur` (`id_U`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB
```

Tables appareil

```
CREATE TABLE IF NOT EXISTS `Jarvis`.`Appareil` (  
  `id_appareil` INT NOT NULL,  
  `Nom_appareil` VARCHAR(45) NULL,  
  `Type_appareil` VARCHAR(45) NULL,  
  PRIMARY KEY (`id_appareil`))  
ENGINE = InnoDB
```

Tables pieces

```
CREATE TABLE IF NOT EXISTS `Jarvis`.`Pieces` (  
  `id_piece` INT NOT NULL,  
  `Nom_piece` VARCHAR(45) NULL,  
  PRIMARY KEY (`id_piece`))  
ENGINE = InnoDB
```

Tables domicile

```
CREATE TABLE IF NOT EXISTS `Jarvis`.`Domicile` (  
  `id_domicile` INT NOT NULL,  
  `Annee_construction` VARCHAR(45) NULL,  
  `Superficie` VARCHAR(45) NULL,  
  PRIMARY KEY (`id_domicile`))  
ENGINE = InnoDB
```

- ❖ Maintenant, on va remplir chaque table avec du champ qui le conviennent, pour faire cela on utilise la commande : **INSERT INTO nom_table(champ,.....champ_n) VALUES (Valeur1,Valeur_n)**

Prenons par exemples pour la table utilisateur :

```
INSERT INTO `jarvis`.`utilisateur`(`id_U`, `nom_U`, `date_naissance_U`, `sexe_U`,  
  `mot_de_passe_U`, `email_U`)  
VALUES('1', 'Rakoto', '1990-02-16', 'H', 'Rkt09', 'rakoto@gmail.com');
```

CREATION DE TRIGGERS

- ❖ Maintenant, on doit procéder à la création de Views

CREATE VIEWS

Views Synthetisant Utilisateur

```
USE `jarvis`;  
CREATE jarvis.`synthetiser_UTILISATEUR` AS  
SELECT * FROM utilisateur
```

Views Synthetisant Membre

```
USE `jarvis`;  
CREATE VIEW jarvis.`synthetiser_membres` AS  
SELECT id_membres, nom_U, date_naissance_U, sexe_U, mot_de_passe_U, email_U  
FROM membres, utilisateur  
WHERE utilisateur.id_U = membres.Utilisateur_id_U;
```

Views Synthetisant Proprietaire

```
USE `jarvis`;  
CREATE VIEW jarvis.`synthetiser_proprietaire` AS  
SELECT id_proprietaire, nom_U, date_naissance_U, sexe_U, mot_de_passe_U, email_U  
FROM proprietaire, utilisateur  
WHERE utilisateur.id_U = proprietaire.Utilisateur_id_U;
```

Views Synthetisant Co-propietaire

```
USE `jarvis`;  
CREATE VIEW jarvis.`synthetiser_co_proprietaire` AS  
SELECT id_Co_proprietaire, nom_U, date_naissance_U, sexe_U, mot_de_passe_U, email_U  
FROM co_proprietaire, utilisateur  
WHERE utilisateur.id_U = id_U;
```

❖ On veut lister toute les tables dans nos bases données.

LISTES DES TOUTES PIECES D'UN DOMICILE

```
SELECT Nom_piece, id_domicile FROM pieces, domicile  
WHERE id_domicile = id_piece;
```

LISTES DES TOUS LES APPAREILS D'UN DOMICILE

```
SELECT Nom_appareil, id_domicile FROM appareils, domicile  
WHERE id_appareil = id_domicile;
```

LISTES DES TOUTES UTILISATEURS D'UN DOMICILE

```
SELECT nom_U, id_domicile FROM utilisateur, domicile  
WHERE id_U = id_domicile;
```

LISTES DES TOUS LES APPAREILS D'UN PIECES

```
SELECT Nom_appareil, id_piece FROM appareil, pieces  
WHERE id_appareil = id_piece;
```

LISTES DES TOUTES DES INVITATIONS EN ATTENTE

LISTES DES PROPRIETAIRES ET CO-PROPRIETAIRE

```
SELECT nom_U, id_proprietaire, id_Co_proprietaire from utilisateur, proprietaire, co_proprietaire  
where id_U = proprietaire.Utilisateur_id_U OR id_U = co_proprietaire.Utilisateur_id_U;
```