



SAYNA

Création d'une view synthétisant user, membre, copropriétaire et propriétaire :

```
USE `jarvis2`;
```

```
CREATE OR REPLACE VIEW `liste_utilisateurs` AS
```

```
SELECT * FROM jarvis2.utilisateur;
```

```
USE `jarvis2`;
```

```
CREATE OR REPLACE VIEW `liste_membre` AS
```

```
SELECT id_u, id_m, prenom_u, nom_u, date_naissance_u, email_u, sexe_u, mdp_u FROM  
jarvis2.membre, jarvis2.utilisateur
```

```
WHERE utilisateur_id_u=id_u;
```

```
USE `jarvis2`;
```

```
CREATE OR REPLACE VIEW `liste_copropriétaires` AS
```

```
SELECT id_u, id_copro, prenom_u, nom_u, date_naissance_u, email_u, sexe_u, mdp_u FROM  
jarvis2.coproprietaire, jarvis2.utilisateur
```

```
WHERE utilisateur_id_u=id_u;
```

```
USE `jarvis2`;
```

```
CREATE OR REPLACE VIEW `liste_propriétaire` AS
```

```
SELECT id_u, id_pro, prenom_u, nom_u, date_naissance_u, email_u, sexe_u, mdp_u FROM  
jarvis2.proprietaire, jarvis2.utilisateur
```

```
WHERE utilisateur_id_u=id_u;
```

Création d'un trigger se déclenchant à l'acceptation d'une invitation, supprimant le membre de tout domicile et l'ajoutant dans le nouveau domicile (en cas de suppression d'un propriétaire, le copropriétaire ou le membre devient propriétaire, sinon le domicile est supprimé en cascade) :

```
CREATE DEFINER=`root`@`localhost` TRIGGER `inviter_AFTER_UPDATE` AFTER UPDATE ON `inviter`  
FOR EACH ROW BEGIN
```

```
IF (new.statutinvitation = 1)
```

```
THEN DELETE FROM jarvis2.contient
```

WHERE contient.utilisateur_id_u = old.utilisateur_id_u;

END IF;

END

○ **Lister toutes les pièces d'un domicile :**

SELECT id_domicile, nom_piece FROM domicile, piece

where contenir_piece_id_piece=id_piece;

○ **Lister tous les appareils d'un domicile :**

SELECT id_domicile, nom_appareil, type_appareil FROM domicile, appareils

where id_appareil=contenir_piece_appareils_id_appareil;

○ **Lister tous les utilisateurs d'un domicile :**

SELECT domicile_id_domicile, id_u, prenom_u, nom_u FROM jarvis2.contient, utilisateur

where utilisateur_id_u=id_u;

○ **Lister les propriétaires et copropriétaires :**

SELECT id_pro, id_u, prenom_u, nom_u FROM proprietaire, utilisateur

where utilisateur_id_u=id_u ;

SELECT id_copro, id_u, prenom_u, nom_u FROM coproprietaire, utilisateur

where utilisateur_id_u=id_u ;

○ **Lister tous les appareils d'une pièce :**

SELECT nom_appareil, type_appareil, nom_piece FROM piece, appareils

where id_appareil=appareils_id_appareil;

○ **Lister toutes les invitations en attentes :**

SELECT id_u, prenom_u, nom_u, lien_parente, statutinvitation FROM inviter, utilisateur

where utilisateur_id_u=id_u;