

# TP PostgreSQL n°3

## - Requêtes avancées et transactions -

Il est conseillé de travailler le plus possible avec des fichiers sql plutôt qu'avec l'éditeur de PostgreSQL .

### PostgreSQL et psql

Connectez-vous à la base de données (tp\_intro ou autre bdd déjà existante).

Commandes de base. Les principales commandes de psql sont les suivantes :

- \? : Liste des commandes psql,
- \h : liste des instructions SQL,
- \h <une\_instruction> : description de l'instruction SQL <une\_instruction>.

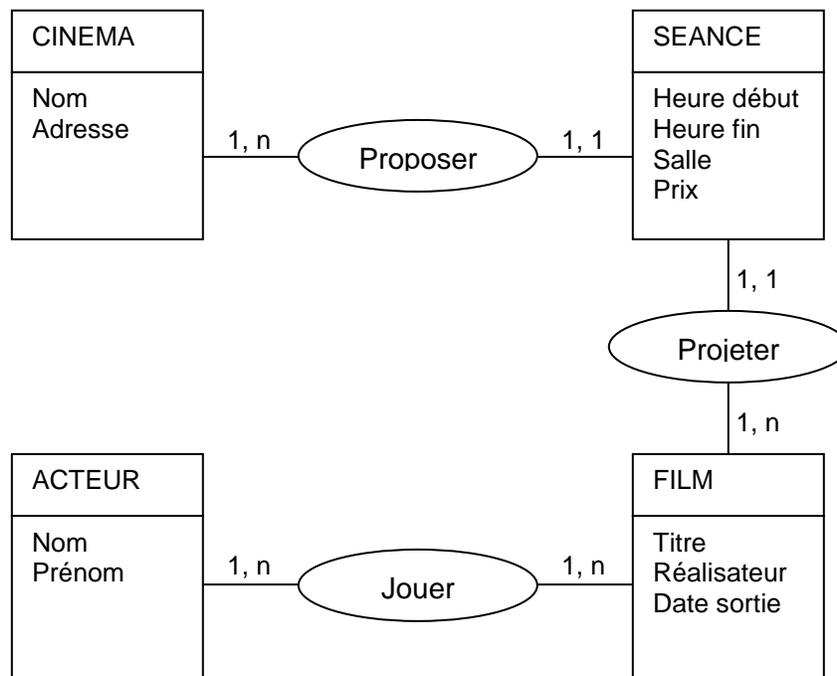
Essayez avec l'instruction CREATE TABLE,

- \d : liste des tables (ou relations, cf. partie 2),
- \d <nom\_table> : description de la table <nom\_table>,
- \i <chemin/nom\_fichier\_script.sql> : exécution d'un fichier de script SQL,
- \o <chemin/nom\_fichier\_resultat.sql> : écriture des résultats des instructions dans le fichier passé en paramètre,
- \o : retour à un affichage à l'écran,.
- \q : quitter psql.

Retenez ces commandes, elles seront indispensables pour les TP.

### 1°) Modèle Entité/Association

Le fichier cinema.sql contient la définition ainsi que des données de test d'une base de données. Cette dernière a été conçue à partir du modèle Entité – Associations suivant :



Charger le fichier /users/prof/fernandes/data/cinema.sql contenant la description des tables et les données. Noter la présence d'une cinquième table permettant de faire la gestion de la relation N-N entre les entités **Acteur** et **Film**.

Les tables suivantes sont donc créées :

- Cinema (idcine, nom, adresse)
- Film (idfilm, titre, realisateur, datesortie)
- Seance (idseance, heuredbt, heurefin, numsalle, **film,cine**,prix)
- Acteur (idacteur, nom, prenom)
- Casting (idacteur, idfilm, personnage)

### Quelle est la commande permettant de voir la liste des tables créées ?

Ex Oracle : En lançant la requête suivante, vous pouvez vérifier que les tables ont bien été créées dans le catalogue : `SELECT * FROM cat`

### Quelle est la commande permettant de vérifier la présence des contraintes dans une table ?

Exemple Oracle : La commande suivante permet de vérifier la présence de contraintes sur les tables :

```
SELECT constraint_name, constraint_type, table_name FROM user_constraints;
```

## 2°) Requêtes partie 1

Quelles requêtes permettent de répondre aux questions suivantes ? Sauvegarder l'ensemble des requêtes dans un fichier tp3-1.sql

1. Afficher toutes les informations propres aux cinémas
2. Afficher le titre de tous les films
3. Afficher le titre de tous les films par ordre alphabétique
4. Afficher le titre des films sortis en 1995 ou après
5. Afficher le titre des films sortis entre 1980 et 1995 par ordre alphabétique inverse
6. Affiche le titre des films, et leur date de sortie, pour les films antérieur à 1999 et dont le titre contient la chaîne 'in'. Renommer le titre de la colonne datesortie en Sortie (opérateur AS).
7. Afficher le nom et le prénom de tous les acteurs
8. Afficher le nom des acteurs et des cinémas
9. Afficher le nom et le prénom des acteurs dont le nom commence par D
10. Afficher le nom (en majuscule) des acteurs et l'initiale de leur prénom suivie d'un point, c'est à dire sous le format NOM P. (opérateurs UPPER et CONCAT). Renommer le nom de la colonne en Nom
11. Afficher les heures de début et de fin de chaque séance sous le format hh-mm
12. Afficher les heures de début et de fin de chaque séance commençant à ou après 19h00 sous le format hh-mm
13. Afficher les heures de début et de fin de chaque séance commençant à ou après 19h00 sous le format hh-mm, en renommant les colonnes Debut et Fin