

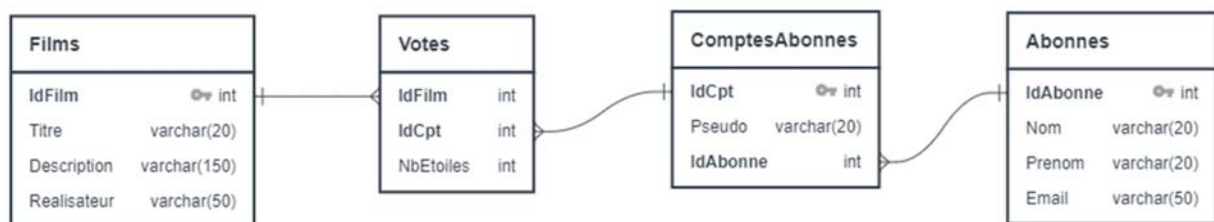
### EXERCICE 5 : Vidéos à la demande (4 points)

Cet exercice traite principalement du thème « traitement de données en tables et bases de données ».






Les fournisseurs de VOD (vidéos à la demande) permettent à leurs abonnés d'avoir plusieurs comptes (afin que chaque membre de la famille puisse avoir son espace client rattaché à un compte unique pour la famille). Ils proposent également un service de votes et de conseils.

#### Partie A : Modèle relationnel

Voici le modèle relationnel qui sera utilisé dans cet exercice :



Lecture :

- un symbole  identifie une clé primaire
- un trait  entre deux attributs indique qu'ils doivent partager les mêmes valeurs. Ces valeurs sont prises de manière unique dans la relation du côté  et zéro, une ou plusieurs fois du côté  à condition d'être présente du côté .

1. Donner les clés primaires des relations *Films* et *Abonnes*.
2. Le code SQL suivant a été utilisé pour créer la relation *Films*. Donner le domaine (c'est-à-dire le type) des attributs *IdFilm* et *Description*.

```
CREATE TABLE Films (  
  IdFilm INT AUTO_INCREMENT PRIMARY KEY,  
  Titre VARCHAR(20),  
  Description VARCHAR(150),  
  Realisateur VARCHAR(50)  
);
```

3. Préciser la clé primaire et la clé étrangère de la relation *ComptesAbonnes*.

L'entreprise gérant le service de VOD demande une adaptation du modèle relationnel.

4. Elle souhaite pouvoir stocker les acteurs principaux de chaque film. Préciser les modifications à apporter.

5. Elle souhaite pouvoir associer une tranche d'âge (-12 ans, 13-15 ans, 16-18 ans, +18 ans) à chacun des comptes d'un abonné et à attribuer, à chaque film, un âge minimum à avoir avant de le visionner. Préciser les modifications à apporter.

## Partie B : Requêtes SQL

1. Écrire une requête SQL permettant de récupérer l'identifiant et le pseudo de tous les comptes rattachés à l'abonné 237

2. Expliquer ce que fait la requête SQL suivante.

```
SELECT AVG(NbEtoiles)
FROM Votes
WHERE IdFilm = 1542;
```

3. Expliquer ce que fait la requête SQL suivante.

```
SELECT u.IdFilm, u.Titre, v.NbEtoiles
FROM Films as u JOIN Votes as v
ON u.IdFilm = v.IdFilm
WHERE v.IdCpt = 508
ORDER BY v.NbEtoiles DESC;
```

4. Écrire une requête SQL permettant de modifier le pseudo du compte 508 pour lui attribuer la valeur « Champion ».

## Partie C : Conseils de films

Dans cette partie, nous utiliserons un module Python dont l'interface est fournie ci-dessous.

Module ConsultationBaseVOD.py	
podiumCompte(IdCpt)	Prend en paramètre un identifiant de compte et renvoie la liste des films les mieux notés par l'utilisateur de ce compte ( <i>au maximum 10 films</i> ) par ordre décroissant de nombre d'étoiles.
spectateurs(listeFilms)	Prend en paramètre une liste de films et renvoie une liste contenant les identifiants des comptes sur lesquels tous ces films ont été visionnés.
nbEtoiles(IdFilm, IdCpt)	Prend en paramètre un identifiant de film et un identifiant de compte. Renvoie le nombre d'étoiles attribuées à ce film par cet utilisateur.

1. Expliquer ce que fait la fonction ci-dessous.

```
def distance(IdCpt1, IdCpt2, listeFilms):  
    assert (type(listeFilms) is list) and (len(listeFilms) !=  
0)  
    somme_ecarts = 0  
    for film in listeFilms:  
        somme_ecarts += abs(nbEtoiles(film, IdCpt1) - \  
                           nbEtoiles(film, IdCpt2))  
    return somme_ecarts / len(listeFilms)
```

2. Traduire en Python la fonction `conseilsFilms` suivante qui permet, à partir d'un identifiant de compte, de conseiller des films proches des goûts de l'utilisateur de ce compte.

Fonction `conseilsFilms (IdCpt)`

Créer une liste vide nommée « conseils »

Récupérer la liste des films les mieux notés par l'utilisateur du compte passé en paramètre

Récupérer la liste des spectateurs ayant visionné ces films

Pour chacun de ces spectateurs

    Si la distance avec l'utilisateur du compte passé en paramètre est inférieure à 10

        Ajouter à la liste « conseils » les films préférés de ce spectateur (maximum 3)

Renvoyer la liste « conseils »

Remarque : Les répétitions sont autorisées dans la liste « conseils ».