

Base de données : Série n°2

Exercice

La base de données **films** est décrite par la représentation graphique illustrée en figure 1 et par les requêtes SQL suivantes :

```
CREATE TABLE acteur (  
    act_id INT AUTO_INCREMENT PRIMARY KEY,  
    act_nom VARCHAR(64) NOT NULL,  
    act_dn DATE);  
  
CREATE TABLE film (  
    film_id INT AUTO_INCREMENT PRIMARY KEY,  
    film_nom VARCHAR(192) NOT NULL,  
    film_annee INT,  
    film_type VARCHAR(32));  
  
CREATE TABLE acteur_film (  
    film_id INT,  
    act_id INT);  
  
CREATE TABLE nationalite_film (  
    film_id INT,  
    film_pays VARCHAR(64));
```

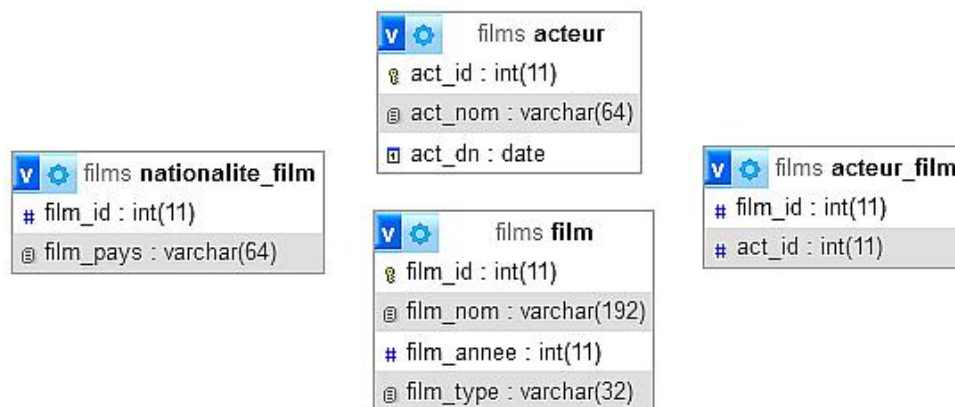


Figure 1: Représentation graphique incomplète de la base de données **films**

Le concepteur de cette base de données a commis plusieurs erreurs :

- La table "**nationalite_film**" est inutile, le champ "**film_pays**" de cette table doit être ajouté à la table "**film**" avant de la supprimer.
- La table "**acteur_film**" ne possède pas de clé primaire.
- Aucune relation entre les tables n'a été définie.
- La table "**acteur_film**" indique qu'un acteur a joué dans un film, cependant, elle n'indique pas quel est le rôle qu'il a joué dans le film.

La base de données contient les données présentées dans le tableau 1.

Table **acteurs**

act_id	act_nom	act_dn
1	Keanu Reeves	1964-09-02
2	Carrie-Anne Moss	1967-08-21
3	Laurence Fishburne	1961-07-30
4	Jada Oinkett Smith	1971-09-18
5	Hugo Weaving	1960-04-04
6	Daniel Bernhardt	1965-08-31
7	Vin Diesel	1967-07-18
8	Paul Walker	1973-09-12
9	Michelle Rodriguez	1978-07-12
10	Jordana Brewster	1980-04-26
11	Matt Schulze	1972-07-03
12	Ian Mc Shane	1942-09-29
13	Lance Reddick	1962-07-07
14	John Leguizamo	1960-07-22

Table **films**

film_id	film_nom	film_annee	film_type
1	Matrix	1999	Science-fiction
2	Fast and Furious	2001	Aventure
3	John Wick	2014	Action

Table **nationalite_film**

film_id	film_pays
1	USA
2	USA
3	USA

Tableau 1: Les données insérées initialement dans la base de données

Travail demandé

1. Créer la base de données **films**. Choisir l'encodage "**utf8_general_ci**".
2. Importer le contenu du fichier "**conception01.sql**" qui contient les requêtes de création des tables et les requêtes d'insertion des données.
3. Ajouter le champ "**film_pays**" à la table "**film**". Ce champ est obligatoire, de type texte, et sa taille maximale est de 64 caractères.
4. Renommer le champ "**film_nom**" en "**film_titre**" tout en conservant le même type et la même taille de champ.
5. Insérer les données du champ "**film_pays**" de la table "**nationalite_film**" dans le champ portant le même nom de la table "**film**". On utilisera pour cela la requête suivante :

```
UPDATE film
SET film_pays = (SELECT film_pays
                  FROM nationalite_film AS nf
                  WHERE nf.film_id = film.film_id);
```

6. Supprimer la table "**nationalite_film**".
7. Définir les champs "**film_id**" et "**act_id**" comme étant la clé primaire de la table "**acteur_film**".
8. Ajouter le champ "**personnage**" à la table "**acteur_film**" pour indiquer le rôle joué par un acteur dans un film. Ce champ est optionnel, de type texte et de taille 48 caractères.
9. Ajouter les relations entre les tables "**acteur_film**", "**film**" et "**acteur**" :

Le champ "**film_id**" est une clé étrangère dans la table "**acteur_film**".

Le champ "**act_id**" est une clé étrangère dans la table "**acteur_film**".

Activer la mise à jour en cascade et la suppression en cascade pour les clés étrangères.

10. Vérifier à l'aide du concepteur de **phpMyAdmin** que la base de données "**films**" possède la représentation graphique de la figure 2.

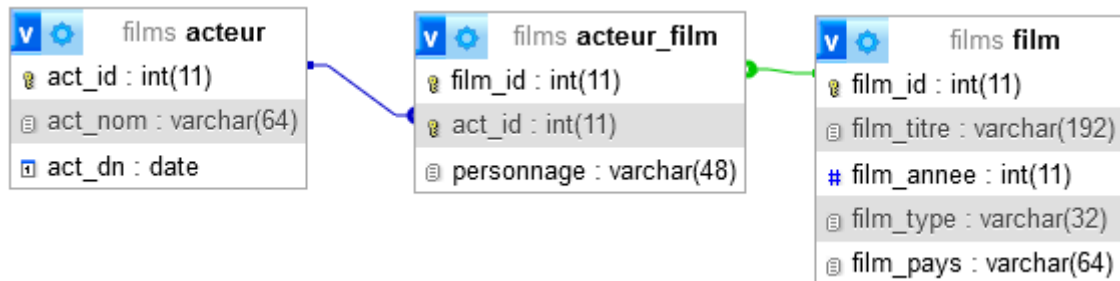


Figure 2: Représentation graphique finale de la base de données *films*

11. Importer les données de la table "**acteur_film**" depuis le fichier "**conception03.sql**".

Table **acteur_film**

film_id	act_id	personnage
1	1	Neo
1	2	Trinity
1	3	Morpheus
1	4	Niobe
1	5	Agent Smith
1	6	Agent Johnson
2	7	Dominic Toretto
2	8	Brian O'Conner
2	9	Letty Ortiz
2	10	Mia Toretto
2	11	Matt Shulze
3	1	John Wick
3	3	Bowery King
3	12	Winston
3	13	Charon
3	14	Aurelio

Tableau 2: Les données insérées dans la table *acteur_film*

12. Écrire les requêtes de sélection suivantes :
- Afficher la liste des informations des acteurs en ordre alphabétique.
 - Afficher la liste des films (titre, année) en ordre chronologique décroissant.

- c) Afficher la liste des acteurs (nom, date de naissance) nés dans les années 70 du plus jeune au plus vieux.
- d) Afficher la liste des films et des personnages interprétés par les acteurs dans ces films (acteur, titre film, année, personnage) ordonner par année, puis par titre et nom d'acteur.
- e) Déterminer qui est l'acteur qui a interprété "Toretto" dans le film "Fast and Furious" et quel était son âge à la date du film.
- f) Retrouver l'acteur le plus âgé dans chaque film. Ordonner par l'année du film puis par le nom de l'acteur.
- g) Déterminer le noms de l'acteur qui a interprété le rôle de "Neo" dans le film "Matrix" produit en 1999 ainsi que la liste de tous les films dans lesquels il a joué.

Réponses

```
-- Q3
ALTER TABLE film
ADD COLUMN film_pays VARCHAR(64) NOT NULL;

-- Q4
ALTER TABLE film
CHANGE COLUMN film_nom film_titre VARCHAR(192) NOT NULL;

-- Q5
UPDATE film
SET film_pays = (SELECT film_pays
                  FROM nationalite_film AS nf
                  WHERE nf.film_id = film.film_id);

-- Q6
DROP TABLE nationalite_film;

-- Q7
ALTER TABLE acteur_film
ADD CONSTRAINT PRIMARY KEY (film_id, act_id);

-- Q8
ALTER TABLE acteur_film
ADD COLUMN personnage VARCHAR(48);

-- Q9
ALTER TABLE acteur_film
ADD CONSTRAINT FOREIGN KEY (film_id)
    REFERENCES film (film_id)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
ADD CONSTRAINT FOREIGN KEY (act_id)
    REFERENCES acteur (act_id)
    ON DELETE CASCADE
    ON UPDATE CASCADE;
```

```
-- Q12.a.
-- Liste des information des Acteurs en ordre alphabétique
SELECT *
FROM acteur
ORDER BY act_nom;

-- Q12.b (projection)
-- Liste des films (titre, annee) en ordre chronologique décroissant
SELECT film_titre, film_annee
FROM film
ORDER BY film_annee DESC;

-- Q12.c. (projection & sélection)
-- Liste des acteurs (nom, date de naissance) nés dans les années 70
-- du plus jeune au plus vieux.
SELECT act_nom, act_dn
FROM acteur
WHERE YEAR(act_dn) BETWEEN 1970 AND 1979
ORDER BY act_dn DESC;

-- Q12.d. (Projection + Jointure)
-- Liste des films et des personnages interprétés par les acteurs dans ces
-- films (acteur, titre film, annee, personnage) ordonner par année,
-- puis par titre et nom d'acteur.
SELECT act_nom, film_titre, film_annee, personnage
FROM acteur AS a, film AS f, acteur_film AS af
WHERE af.act_id = a.act_id AND af.film_id = f.film_id
ORDER BY film_annee, film_titre, act_nom;

-- Q13.e. (Projection + Jointure + Sélection)
-- Déterminer qui est l'acteur qui a interprété "Toretto" dans
-- le film "Fast and Furious" et quel était son age à la date du film.
SELECT act_nom, (film_annee - YEAR(act_dn)) AS Age
FROM acteur AS a, film AS f, acteur_film AS af
WHERE af.act_id = a.act_id AND af.film_id = f.film_id AND
      act_nom LIKE '%Toretto%' AND
      film_titre = 'Fast and Furious';
```

```
-- Q13.f. (Projection + Jointure + Sous-requête)
-- Retrouver l'acteur le plus âgé dans chaque film.
-- Ordonner par l'année du film puis par le nom de l'acteur.
SELECT act_nom, film_titre, film_annee
FROM acteur AS a, film AS f, acteur_film AS af
WHERE af.act_id = a.act_id AND af.film_id = f.film_id AND
      (f.film_id, act_dn) IN (
        SELECT film_id, MIN(act_dn)
        FROM acteur AS a, acteur_film AS af
        WHERE af.act_id = a.act_id
        GROUP BY film_id
      )
ORDER BY film_annee, act_nom;

-- Q14.g.
-- Déterminer le noms de l'acteur qui a interprété le rôle de "Neo"
-- dans le film "Matrix" produit en 1999 ainsi que la liste
-- de tous les films dans lesquels il a joué.
SELECT act_nom, film_titre, film_annee, personnage
FROM acteur AS a, film AS f, acteur_film AS af
WHERE af.act_id = a.act_id AND af.film_id = f.film_id AND
      a.act_id IN (
        SELECT act_id
        FROM acteur_film AS af, film AS f
        WHERE af.film_id = f.film_id AND
              film_annee = 1999 AND
              personnage = 'Neo'
      );
```