

Projet BDDr - 8s G2

by Raphaël CANIN, Amanda NGUYEN, Romain SAVINO, Yann SOBGUI & Rémy XIA

Problématique

Créer un catalogue de films permettant la recherche selon son genre, la notation, la durée...

Besoins

Recherche/classement simple et rapide de films correspondant à des critères plus ou moins précis (langue, note, genre...)

Notre base de données

<https://www.imdb.com/interfaces/>

<https://www.kaggle.com/stefanoleone992/imdb-extensive-dataset>

Modèle EA

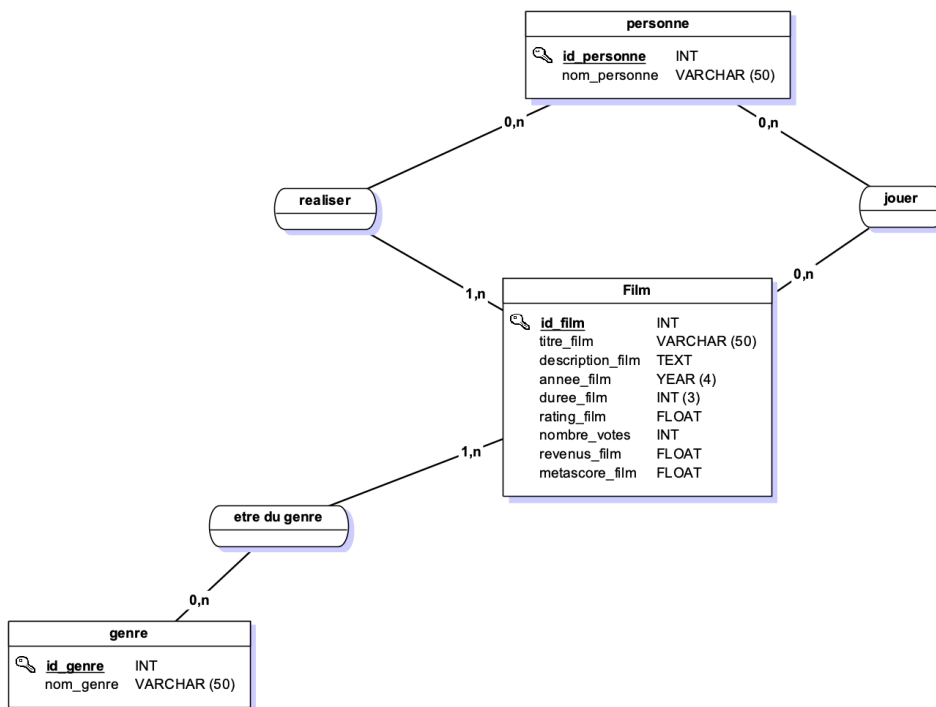
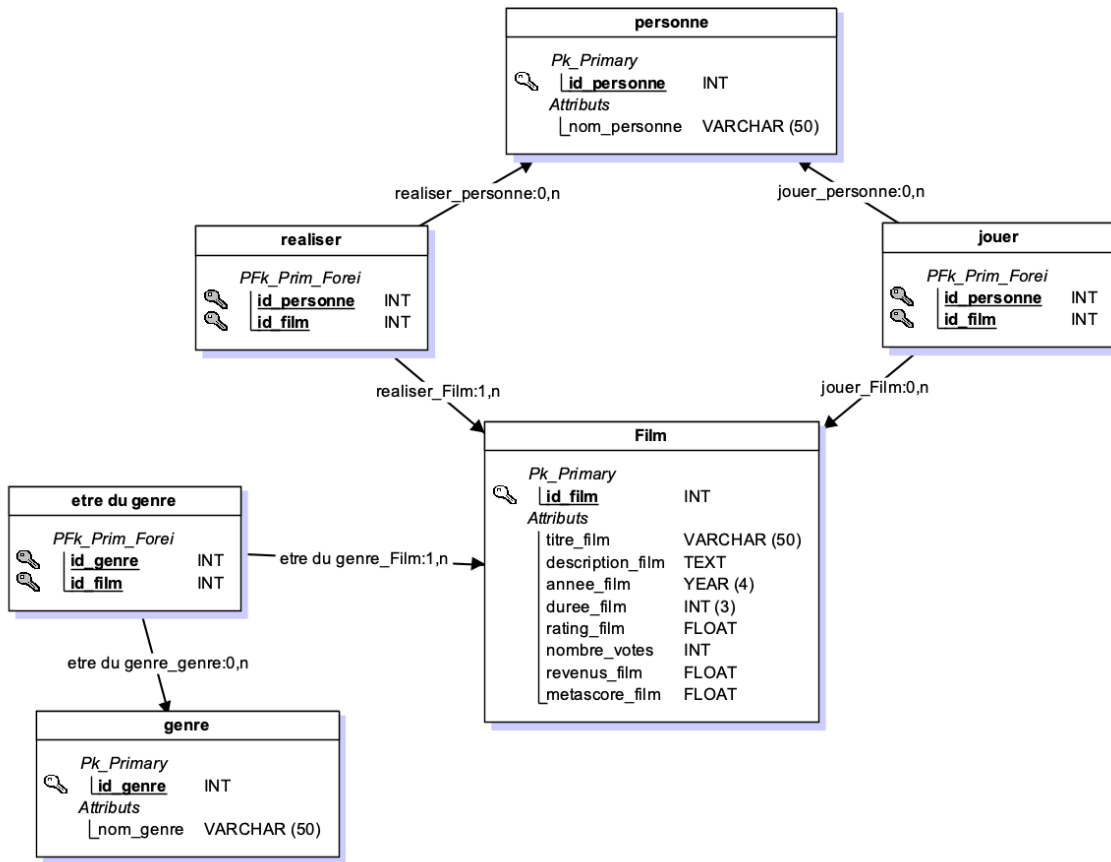


Schéma relationnel



Création SQL

```
#-----
#   Script MySQL.
#-----
```

```
#-----
# Table: genre
#-----
```

```
CREATE TABLE genre(
    id_genre Int NOT NULL ,
    nom_genre Varchar (50) NOT NULL
    ,CONSTRAINT genre_PK PRIMARY KEY (id_genre)
)ENGINE=InnoDB;
```

```
#-----
# Table: Film
#-----
```

```

CREATE TABLE film(
    id_film      Int NOT NULL ,
    titre_film   Varchar (50) NOT NULL ,
    description_film Text NOT NULL ,
    annee_film   Year NOT NULL ,
    duree_film   Int NOT NULL ,
    rating_film  Float NOT NULL ,
    nombre_votes Int NOT NULL ,
    revenus_film Float NOT NULL ,
    metascore_film Float NOT NULL
    ,CONSTRAINT film_PK PRIMARY KEY (id_film)
)ENGINE=InnoDB;

```

```

#-----
# Table: personne
#-----

```

```

CREATE TABLE personne(
    id_personne Int NOT NULL ,
    nom_personne Varchar (50) NOT NULL
    ,CONSTRAINT personne_PK PRIMARY KEY (id_personne)
)ENGINE=InnoDB;

```

```

#-----
# Table: etre du genre
#-----

```

```

CREATE TABLE etre_du_genre(
    id_genre Int NOT NULL ,
    id_film  Int NOT NULL
    ,CONSTRAINT etre_du_genre_PK PRIMARY KEY (id_genre,id_film)

    ,CONSTRAINT etre_du_genre_genre_FK FOREIGN KEY (id_genre) REFERENCES
genre(id_genre)
    ,CONSTRAINT etre_du_genre_film0_FK FOREIGN KEY (id_film) REFERENCES
film(id_film)
)ENGINE=InnoDB;

```

```

#-----
# Table: realiser
#-----

```

```

CREATE TABLE realiser(
    id_personne Int NOT NULL ,

```

```

        id_film    Int NOT NULL
        ,CONSTRAINT realiser_PK PRIMARY KEY (id_personne,id_film)

        ,CONSTRAINT realiser_personne_FK FOREIGN KEY (id_personne) REFERENCES
personne(id_personne)
        ,CONSTRAINT realiser_film0_FK FOREIGN KEY (id_film) REFERENCES
film(id_film)
    )ENGINE=InnoDB;

```

```

#-----
# Table: jouer
#-----

```

```

CREATE TABLE jouer(
    id_personne Int NOT NULL ,
    id_film    Int NOT NULL
    ,CONSTRAINT jouer_PK PRIMARY KEY (id_personne,id_film)

    ,CONSTRAINT jouer_personne_FK FOREIGN KEY (id_personne) REFERENCES
personne(id_personne)
    ,CONSTRAINT jouer_film0_FK FOREIGN KEY (id_film) REFERENCES film(id_film)
)ENGINE=InnoDB;

```

Exemple SQL :

Tous les films réalisés par Christopher Nolan :

```

SELECT * FROM film JOIN realiser ON film.id_film=realiser.id_film JOIN personne ON
realiser.id_personne=personne.id_personne WHERE personne.nom_personne="Christopher
Nolan"

```

Tous les films réalisés par AGI (+ gestion des accents):

```

SELECT * FROM film JOIN realiser ON film.id_film=realiser.id_film JOIN personne ON
realiser.id_personne=personne.id_personne WHERE
upper(personne.nom_personne)="ALEJANDRO GONZALEZ INARRITU"
Ensemble des personnes ayant été acteur et réalisateur:
SELECT DISTINCT personne.nom_personne FROM personne JOIN jouer ON
personne.id_personne=jouer.id_personne JOIN realiser ON
jouer.id_personne=realiser.id_personne WHERE jouer.id_personne=realiser.id_personne

```

Genre:

```

SELECT * FROM film JOIN etre_du_genre ON film.id_film=etre_du_genre.id_film JOIN
genre ON genre.id_genre=etre_du_genre.id_genre WHERE nom_genre="action"

```

Double Genre:

```
SELECT * FROM film
```

```
JOIN etre_du_genre ON film.id_film=etre_du_genre.id_film
```

```
JOIN genre ON genre.id_genre=etre_du_genre.id_genre
```

```
WHERE nom_genre="action"
```

```
AND film.id_film IN
```

```
(SELECT film.id_film FROM film
```

```
JOIN etre_du_genre ON film.id_film=etre_du_genre.id_film
```

```
JOIN genre ON genre.id_genre=etre_du_genre.id_genre
```

```
WHERE nom_genre="aventure")
```