

Zachary Tsouprakos  
Lab 8  
Database Management

----- Rachel Ulicni and I collaborated and shared ideas for this lab -----

### **Functional Dependencies**

People Table:

PID → FirstName, LastName, Address

Spouses:

PID → SpouseName

Actors:

PID → DOB, Hair\_Color, Eye\_Color, Height, Weight, Favorite\_Color,  
Guild\_Anniversary\_Date)

ActorsInMovie:

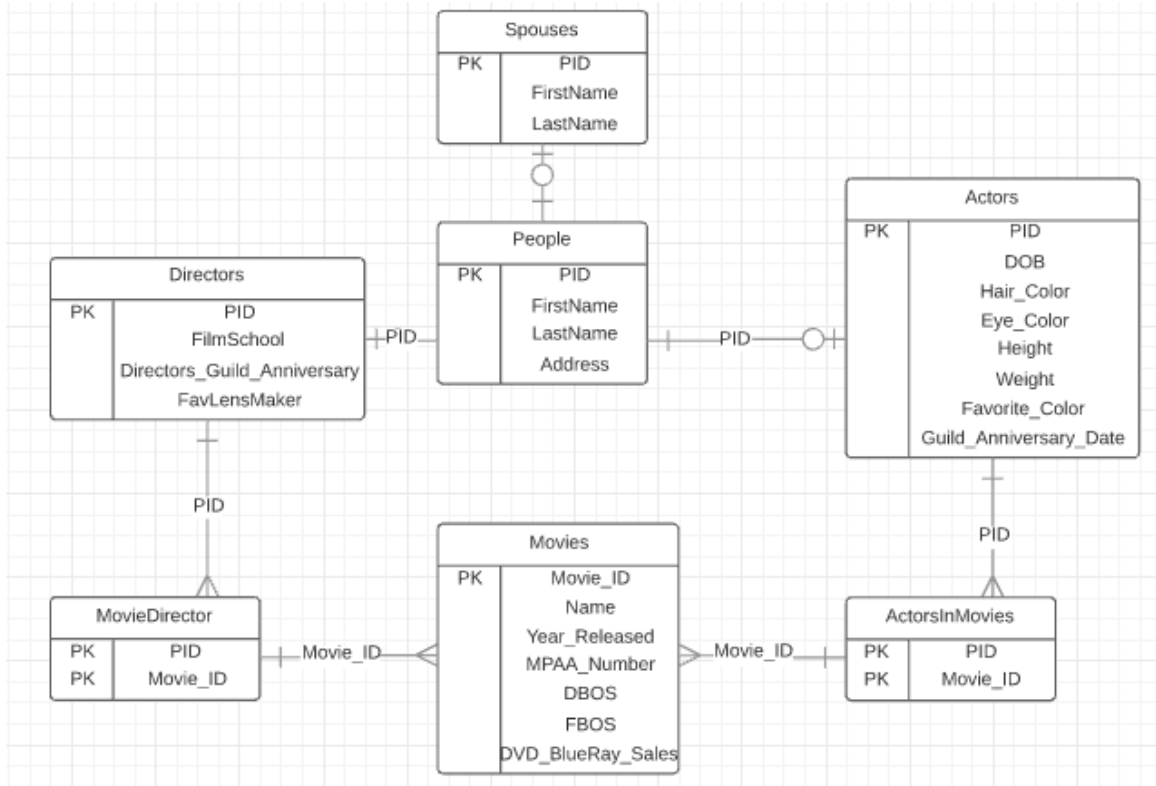
(PID, Movie\_ID) →

MovieDirectors:

(PID, Movie\_ID) →

Movies:

(Movie\_ID) → Movie\_ID, Name, Year\_Released, MPAA\_Number, DBOS, FBOS,  
DVD\_BlueRay\_Sales,



```

DROP TABLE IF EXISTS Actors ;
DROP TABLE IF EXISTS ActorsInMovie ;
DROP TABLE IF EXISTS Movies ;
DROP TABLE IF EXISTS MovieDirectors ;
DROP TABLE IF EXISTS Directors ;
DROP TABLE IF EXISTS Spouses ;
DROP TABLE IF EXISTS People ;
  
```

```

SELECT * FROM information_schema.tables ;
  
```

```

DROP TABLE IF EXISTS actorsinmovie ;
  
```

```

CREATE TABLE IF NOT EXISTS ActorsInMovie (
    Movie_ID INT NOT NULL,
    PID INT NOT NULL,
    PRIMARY KEY (Movie_ID, PID)
);
  
```

```

CREATE TABLE IF NOT EXISTS Actors (
    PID INT NOT NULL,
  
```

```
DOB DATE NOT NULL,  
Hair_Color TEXT,  
Eye_Color TEXT,  
Height int not null,  
WEIGHT INT not null,  
Favorite_Color TEXT not null,  
Guild_Anniversary_Date DATE,  
Primary KEY (PID)  
);
```

```
CREATE TABLE IF NOT EXISTS Movies (  
Movie_ID INT NOT NULL,  
NAME VARCHAR NOT NULL,  
YEAR_Released INT NOT NULL,  
MPAA_Number VARCHAR,  
DBOS INT,  
FBOS INT,  
DVD_BlueRay_Sales INT,  
PRIMARY KEY (Movie_ID)  
);
```

```
CREATE TABLE IF NOT EXISTS MovieDirectors (  
pid INT NOT NULL,  
Movie_ID INT NOT NULL,  
PRIMARY KEY (pid, Movie_ID)  
);
```

```
Create TABLE IF NOT EXISTS Directors (  
pid INT NOT NULL,  
FilmSchool TEXT,  
Directors_Guild_Anniversary date,
```

```
FavLensMaker TEXT,  
Primary KEY (pid)  
);
```

```
CREATE TABLE IF NOT EXISTS People (  
pid INT NOT NULL,  
FirstName TEXT NOT NULL,  
LastName Text NOT NULL,  
Address VarChar,  
Primary KEY (pid)  
);
```

```
Create TABLE IF NOT EXISTS Spouses (  
pid INT NOT NULL,  
FirstName Text Not NULL,  
LastName TEXT NOT NULL,  
Primary KEY (pid)  
);
```

```
INSERT INTO people (PID, FirstName, LastName, Address)  
Values (1, 'Sean', 'Connery', '85 Broadway Avenue'),  
      (2, 'Brad', 'Pitt', '45 Death Star Circle'),  
      (3, 'Guy', 'Hamilton', 'Paris, France'),  
      (4, 'Doug', 'Liman', 'New York, New York')  
;
```

```
INSERT INTO Spouses (PID, FirstName, LastName)  
Values (1, 'Diane', 'Cilento'),  
      (2, 'Angelina', 'Jolie'),  
      (3, 'Naomi', 'Chance')  
;
```

```
INSERT INTO ACTORS (pid, DOB, Hair_Color, Eye_Color, Height, Weight,
Favorite_Color, Guild_Anniversary_Date)
VALUES (1, '8-25-1930', 'White', 'Brown', '74', '185', 'Brown', '6-12-2001'),
      (2, '12-18-1963', 'Brown', 'Blue', '71', '185', 'Green', '5-12-2007')
;
```

```
INSERT INTO Directors (pid, FilmSchool, Directors_Guild_Anniversary,
FavLensMaker)
Values (3, 'The Good One', '2-12-1978', 'Red'),
      (4, 'The Other Good One', '7-14-2007', 'Red');
```

```
Insert INTO MovieDirectors (pid, Movie_ID)
Values (3,1),
      (4,2)
```

```
INSERT INTO Movies (Movie_ID, Name, Year_Released, MPAA_Number, DBOS,
FBOS, DVD_BlueRay_Sales)
Values (1, 'Gold Finger', '1964', 'PG-13', '51081062', '73800000', '500000'),
      (2, 'Mr and Mrs Smith', '2005', 'PG-13', '186336103', '179331224', '500000')
```

```
INSERT INTO ActorsInMovie (pid, Movie_ID)
VALUES(1,1),
      (2,2)
```

```
Query:
SELECT p.pid, p.Firstname, p.LastName
FROM MovieDirectors MD INNER JOIN ActorsInMovie AM on MD.Movie_ID =
AM.Movie_ID
          INNER JOIN People p on MD.pid = p.pid
WHERE AM.pid = '1' ;
```