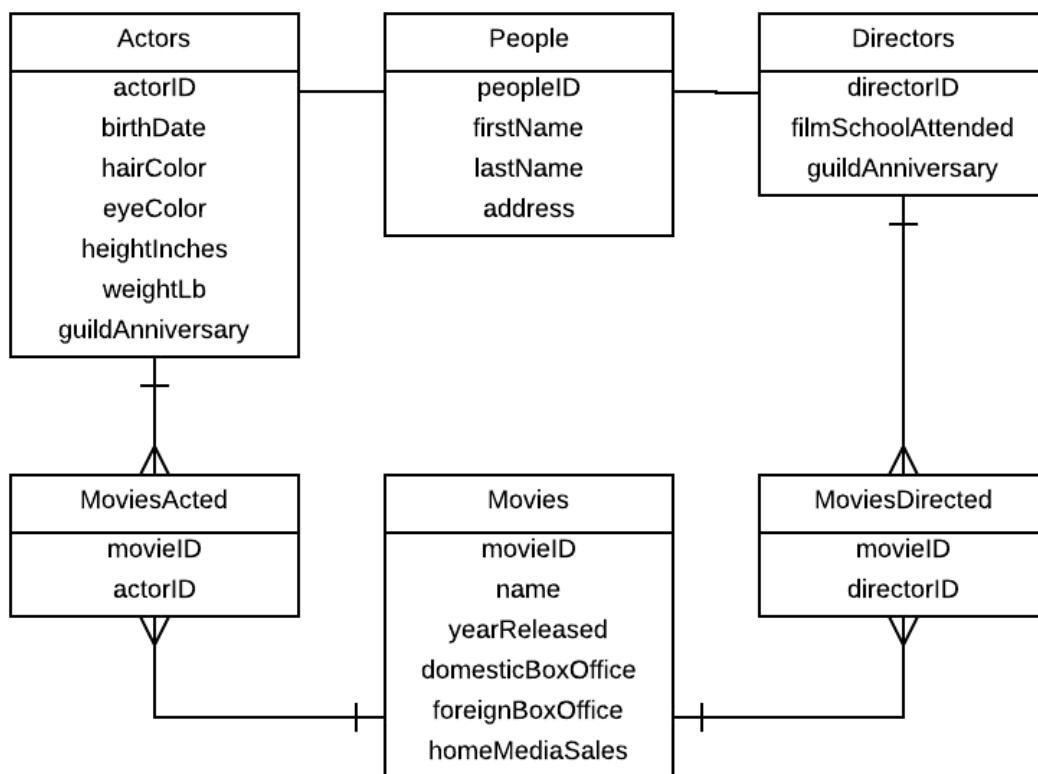Ben DelGiorno

Database Management

CMPT 308L

April 4, 2014

Lab 8 – Normalization Two

1. ER Diagram

2. SQL Create Statements

```sql
drop table if exists People;
create table if not exists People (
      peopleID serial,
      firstName text not null,
      lastName text not null,
      address text not null,
   primary key (peopleID)
);

drop table if exists Actors;
create table if not exists Actors (
      actorID int not null references people (peopleID),
      birthDate date not null,
      hairColor text not null,
      eyeColor text not null,
      heightInches decimal not null,
      weightLb decimal not null,
      guildAnniversary date,
   primary key (actorID)
);

drop table if exists Directors;
create table if not exists Directors (
      directorID int not null references People (peopleID),
      filmSchoolAttended text,
      guildAnniversary date,
   primary key (directorID)
);

drop table if exists Movies;
create table if not exists Movies (
      movieID serial,
      yearReleased int,
      domesticBoxOffice decimal,
      foreignBoxOffice decimal,
      homeMediaSales decimal,
   primary key (movieID)
);

drop table if exists MoviesActed;
create table if not exists MoviesActed (
      movieID int not null references Movies (movieID),
      actorID int not null references Actors (actorID),
   primary key (movieID, directorID)
);

drop table if exists MoviesDirected;
create table if not exists MoviesDirected (
      movieID int not null references Movies (movieID),
      directorID int not null references Directors (directorID),
 primary key (movieID, directorID)
);
```

3. SQL Input Statements

```
-- Inserting Data in People Table
insert into People (firstName, lastName, address)
     values ('Sean', 'Connery', '007 London St.');

insert into People (firstName, lastName, address)
     values ('Tom', 'Hanks', '45 Hollywood Ave.');

insert into People (firstName, lastName, address)
     values ('Brad', 'Pitt', '26 Gilmore St.');

insert into People (firstName, lastName, address)
     values ('Johnny', 'Depp', '13 Pirate Way');

insert into People (firstName, lastName, address)
     values ('Al', 'Pacino', '007 London St.');

insert into People (firstName, lastName, address)
     values ('Steven', 'Spielberg', '123 Movie Rd.');

insert into People (firstName, lastName, address)
     values ('Martin', 'Scorsese', '18 Film St.');

insert into People (firstName, lastName, address)
     values ('Alfred', 'Hitchcock', '7 Bird St.');

insert into People (firstName, lastName, address)
     values ('Francis', 'Coppola', '99 Godfather St.');

-- Inserting Data in Actors table
insert into Actors (actorID, birthDate, hairColor, eyeColor,
                    heightInches, weightLb, guildAnniversary)
      values (1, '1930-08-25', 'White', 'Brown', 74, 200, '1953-11-12');

insert into Actors (actorID, birthDate, hairColor, eyeColor,
                    heightInches, weightLb, guildAnniversary)
      values (2, '1970-05-19', 'Brown', 'blue', 72, 190, '1996-12-08');

insert into Actors (actorID, birthDate, hairColor, eyeColor,
                    heightInches, weightLb, guildAnniversary)
      values (3, '1948-08-25', 'Gray', 'Brown', 70, 180, '1982-07-17');

-- Inserting Data in Directors
insert into Directors (directorsID, filmSchoolAttended, guildAnniversary)
     values (4, 'NYU', '1981-04-19');

insert into Directors (directorsID, filmSchoolAttended, guildAnniversary)
     values (5, 'UCLA', '1996-02-28');

insert into Directors (directorsID, filmSchoolAttended, guildAnniversary)
     values (6, 'Columbia', '1998-11-10');

insert into Directors (directorsID, filmSchoolAttended, guildAnniversary)
     values (7, 'NYU', '2001-03-09');
```

```sql
insert into Directors (directorsID, filmSchoolAttended, guildAnniversary)
    values (8, 'UNC', '2007-06-13');

insert into Directors (directorsID, filmSchoolAttended, guildAnniversary)
    values (9, 'FSU', '1989-05-22');
```

-- **Inserting Data in Movies**
```sql
insert into Movies (name, yearReleased, domesticBoxOffice, foreignBoxOffice,
                    homeMediaSales)
    values ('The Godfather', '1972', 100000000.00, 60000000.00, 38000000.00);

insert into Movies (name, yearReleased, domesticBoxOffice, foreignBoxOffice,
                    homeMediaSales)
    values ('Casablanca', '1942', 80000000.00, 50000000.00,
            14000000.00);

insert into Movies (name, yearReleased, domesticBoxOffice, foreignBoxOffice,
                    homeMediaSales)
    values ('Gone With the Wind', '1939', 90000000.00,
            60000000.00, 38000000.00);

insert into Movies (name, yearReleased, domesticBoxOffice, foreignBoxOffice,
                    homeMediaSales)
    values ('Star Wars', '1977', 200000000.00, 90000000.00, 99000000.00);

insert into Movies (name, yearReleased, domesticBoxOffice, foreignBoxOffice,
                    homeMediaSales)
    values ('Jaws', '1975', 148000000.00, 50000000.00, 68000000.00);

insert into Movies (name, yearReleased, domesticBoxOffice, foreignBoxOffice,
                    homeMediaSales)
    values ('E.T.', '1982', 139000000.00, 87000000.00, 58000000.00);

insert into Movies (name, yearReleased, domesticBoxOffice, foreignBoxOffice,
                    homeMediaSales)
    values ('King Kong', '2005', 161000000.00, 60000000.00, 41000000.00);

insert into Movies (name, yearReleased, domesticBoxOffice, foreignBoxOffice,
                    homeMediaSales)
    values ('World War Z', '2013', 216000000.00, 101000000.00, 67000000.00);
```

-- **Inserting Data in MoviesActed**
```sql
insert into MoviesActed (movieID, actorID)
    values (1, 1);

insert into MoviesActed (movieID, actorID)
    values (2, 1);

insert into MoviesActed (movieID, actorID)
    values (3, 3);

insert into MoviesActed (movieID, actorID)
    values (4, 1);

insert into MoviesActed (movieID, actorID)
    values (5, 2);
```

```
insert into MoviesActed (movieID, actorID)
     values (6, 1);

insert into MoviesActed (movieID, actorID)
     values (7, 2);

insert into MoviesActed (movieID, actorID)
     values (8, 3);

-- Inserting Data in MoviesDirected
insert into MoviesDirected (movieID, directorID)
     values (1, 4);

insert into MoviesDirected (movieID, directorID)
     values (2, 5);

insert into MoviesDirected (movieID, directorID)
     values (3, 6);

insert into MoviesDirected (movieID, directorID)
     values (4, 7);

insert into MoviesDirected (movieID, directorID)
     values (5, 9);

insert into MoviesDirected (movieID, directorID)
     values (6, 8);

insert into MoviesDirected (movieID, directorID)
     values (7, 6);

insert into MoviesDirected (movieID, directorID)
     values (8, 4);

insert into MoviesDirected (movieID, directorID)
     values (9, 5);
```

4. Functional Dependencies
    - People Table
        o peopleID > firstName, lastName, address
    - Actors Table
        o actorID > birthDate, hairColor, eyeColor, heightInches, weightLb, guildAnniversary
    - Directors Table
        o directorID > filmSchoolAttended, guildAnniversary
    - Movies Table
        o movieID > name, yearReleased, domesticBoxOffice, foreignBoxOffice, homeMediaSales

5. SQL Query

```
select distinct p.firstName, p.lastName
from People p, Directors d, Movies m, MoviesDirected md
where d.directorID = md.directorID
  and m.movieID = md.movieID
  and d.directorID = p.peopleID
  and m.movieID in
        (select m.movieID
         from People p, Actors a, Movies m, MoviesActed ma
         where p.peopleID = a.actorID
            and a.actorID = ma.actorID
            and m.movieID = ma.movieID
            and p.firstName = 'Sean'
            and p.lastName = 'Connery'
         );
```