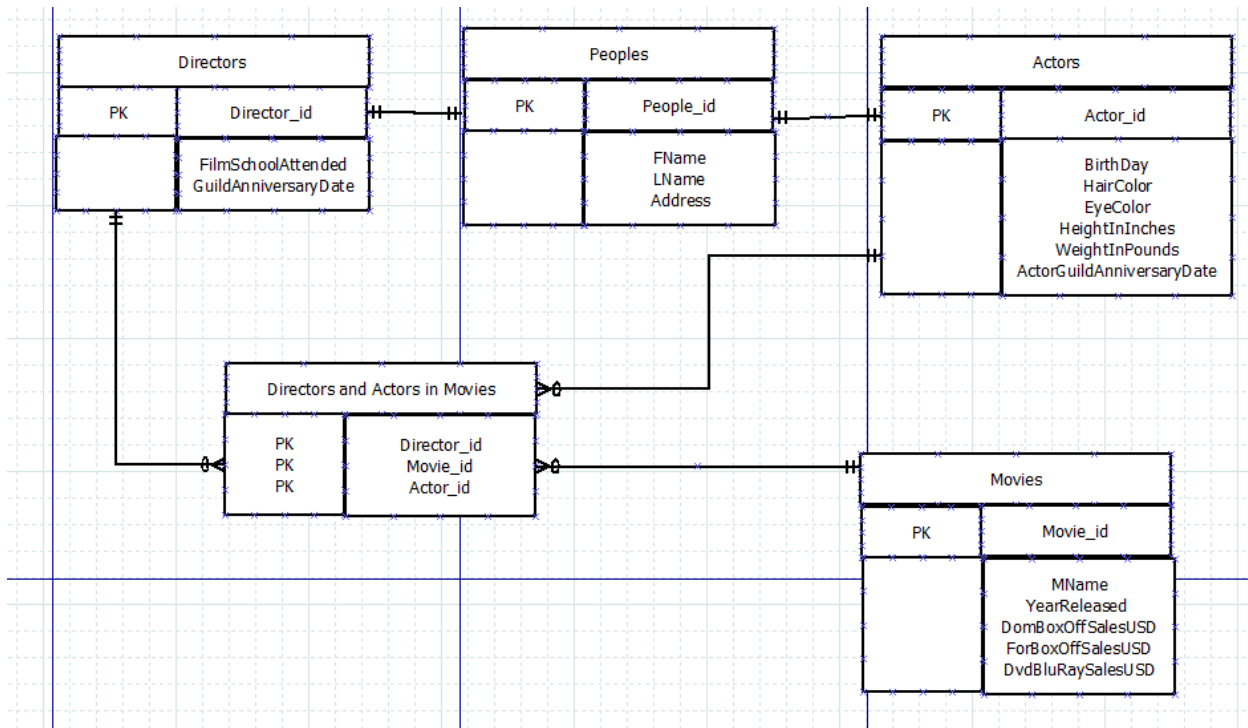


Mason Crane

Normalization 2.

Question 1.



Question 2.

Create statements for Tables --**Peoples Table**

drop table if exists peoples;

create table peoples

(

people_id int not null,

fName char (30) not null,

lName char (30) not null,

Address char (200) not null,

primary key (people_id)

);

--Directors Table

drop table if exists directors;

create table directors

```
(
director_id int not null references Peoples(people_id),
FilmSchoolAttended char (200) not null,
guildAnniversaryDate date not null,
primary key (director_id)
);
```

--Actors Table

drop table if exists actors;

create table actors

```
(
actor_id int not null references Peoples(people_id),
birthDay date not null,
hairColor char (30) not null,
eyeColor char (30) not null,
heightinInches int not null,
weightinPounds int not null,
ActorGuildAnniversaryDate date not null,
primary key (actor_id)
);
```

--Movies Table

drop table if exists movies;

create table movies

```
(
movie_id int not null ,
mName char (50) not null,
yearReleased char (4) not null,
```

```
domBoxOffSalesUSD int not null,  
forBoxOffSalesUSD int not null,  
dvdBluRaySalesUSD int not null,  
primary key (movie_id)  
);
```

--Directors and Actors in Movies Table

```
drop table if exists DirectorsAndActorsInMovies;  
create table DirectorsAndActorsInMovies  
(  
actor_id int not null references actors(actor_id),  
movie_id int not null references movies(movie_id),  
director_id int not null references directors(director_id),  
primary key (movie_id, actor_id, director_id)  
);
```

INSERT STATEMENTS

Peoples Table

```
insert into peoples (people_id, fname, lname, address)  
values ('001', 'Mason', 'Crane', '6220 Deerword Cir North Plymouth MN')
```

```
insert into peoples (people_id, fname, lname, address)  
values ('002', 'Sean', 'Connery', '1401 Main St, Hollywood CA');
```

```
insert into peoples (people_id, fname, lname, address)  
values ('003', 'Jean-Jacques', 'Annaud', '1801 Century Park East 23rd Floor Los Angeles, CA 90067 USA');
```

Directors Table

```
insert into directors (director_id, FilmSchoolAttended, guildAnniversaryDate)  
values ('001', 'New York Film Academy', '11/1/2013')
```

```
insert into directors (director_id, FilmSchoolAttended, guildAnniversaryDate)
values ( '003', 'Institut des Hautes Études Cinématographiques', '11/8/1980')
```

Actors Table

```
insert into actors (actor_id,birthDay, hairColor,eyeColor,heightinInches,
                    weightinPounds, ActorGuildAnniversaryDate)
values ('001', '06/04/1993', 'Dirty Blonde', 'Hazel', '70', '145', '07/06/2011')
```

```
insert into actors (actor_id,birthDay, hairColor,eyeColor,heightinInches,
                    weightinPounds, ActorGuildAnniversaryDate)
values ('002', '9/25/1930', 'brown', 'black', 74, 175, '3/27/2008');
```

Movies Table

```
insert into movies(movie_id, mName, yearReleased, domboxoffsalesUSD, forboxoffsalesUSD, dvdbluRaySalesUSD)
values ( '001', 'The Great Escape', '2009', '100000000', '2000000', '300000000')

insert into movies(movie_id, mName, yearReleased, domboxoffsalesUSD, forboxoffsalesUSD, dvdbluRaySalesUSD)
values ('002', 'The Name Of The Rose', '1986', '7153487', '77000000', '10000000')
```

Directors and Actors in Movies Table

```
insert into DirectorsAndActorsInMovies( actor_id, movie_id, director_id)
values ('001','001','001')

insert into DirectorsAndActorsInMovies( actor_id, movie_id, director_id)
values ('002','002','003')
```

Question 3.

Functional Dependencies

Peoples: People_id → fName , lName, Address

Directors: Director_id → FilmSchoolAttended, GuildAnniversaryDate

Actors: Actor_id → BirthDay, HairColor, EyeColor, HeightInInches, WeightInPounds,
ActorGuildAnniversaryDate

Movies: Movie_id → MName, YearReleased, DomBoxOffSalesUSD, ForBoxOffSalesUSD,
DvdBluRaySalesUSD

DirectorsAndActorsInMovies: (actor_id, director_id, movie_id) →

Question 4.

This is the query to return all the directors with whom actor "Sean Connery" has worked.

```
select distinct p2.fname, p2.lname
from Peoples p1, Peoples p2, Actors a, Directors d, DirectorsAndActorsInMovies daaim
where p1.people_id = a.actor_id
and a.actor_id = daaim.actor_id
and p2.people_id = d.director_id
and d.director_id = daaim.director_id
and p1.fname = 'Sean'
and p1.lname = 'Connery'
```