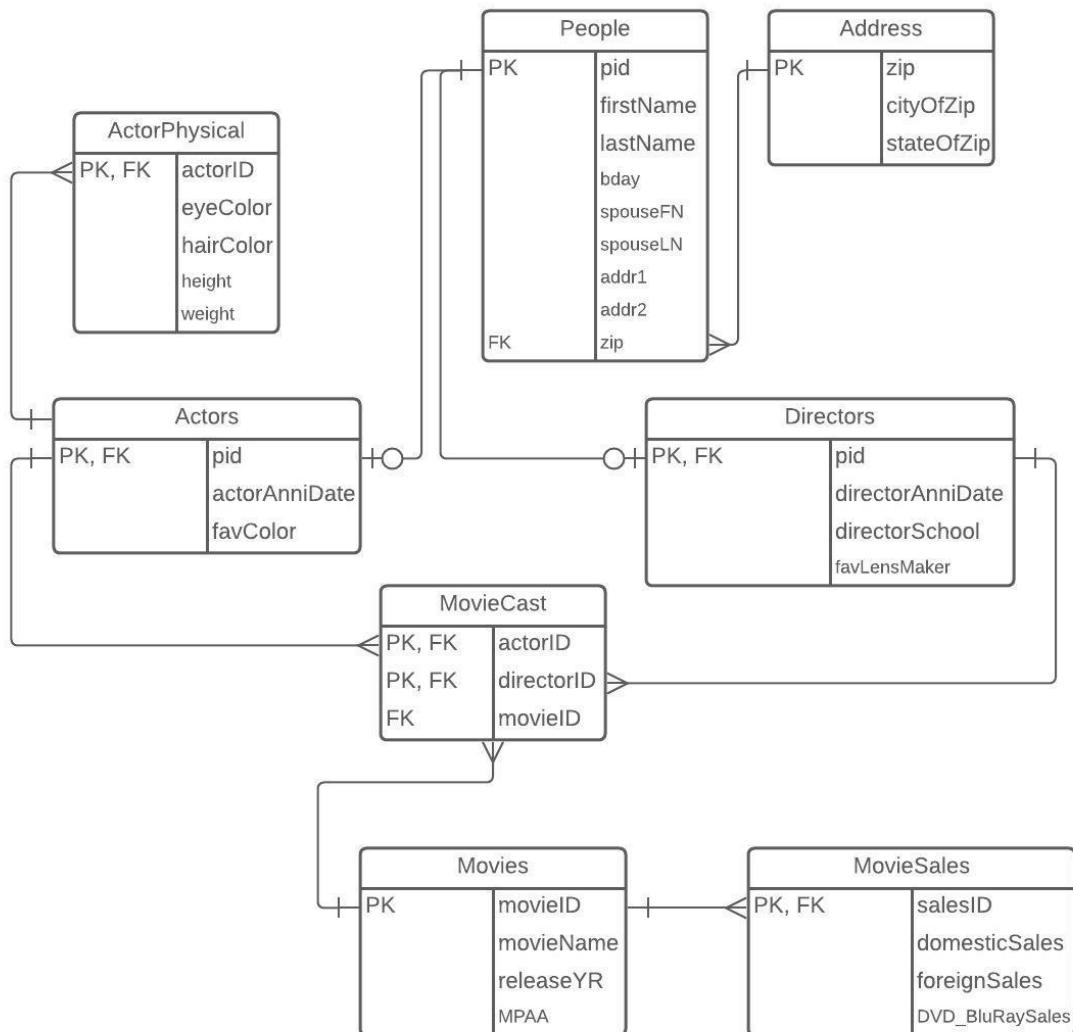Caleb Rogers

Professor Alan Labouseur

November 8th, 2020

Database Systems

<div align="center">Lab 8 – Normalization Two</div>

1)

2)

```
11   -- Create Statements --
12   -- Address --
13   CREATE TABLE Address (
14       zip integer NOT NULL,
15       cityOfZip text,
16       stateOfZip text,
17    PRIMARY KEY(zip)
18   );
19
20   -- People --
21   CREATE TABLE People (
22       pid char(5) NOT NULL,
23       firstName text,
24       lastName text NOT NULL,
25       bday date,
26       spouseFn text,
27       spouseLN text,
28       addr1 text NOT NULL,
29       addr2 text,
30       zip integer NOT NULL references Address(zip),
31    PRIMARY KEY(pid)
32   );
33
34   -- Directors --
35   CREATE TABLE Directors (
36       pid char(5) NOT NULL references People(pid),
37       directorAnniDate date,
38       directorSchool text,
39       favLensMaker text,
40    PRIMARY KEY(pid)
41   );
```

```sql
43  -- Actors --
44  CREATE TABLE Actors (
45      pid char(5) NOT NULL references People(pid),
46      actorAnniDate date,
47      favColor text,
48   PRIMARY KEY(pid)
49  );
50
51  -- ActorPhysical --
52  CREATE TABLE ActorPhysical (
53      actorID char(5) NOT NULL references Actors(pid),
54      eyeColor text,
55      hairColor text,
56      height integer,
57      weight integer,
58   PRIMARY KEY(actorID)
59  );
60
61  -- Movies --
62  CREATE TABLE Movies (
63      movieID char(5) NOT NULL,
64      movieName text,
65      releaseYR integer,
66      MPAA integer,
67   PRIMARY KEY(movieID)
68  );


70  -- MovieCast --
71  CREATE TABLE MovieCast (
72      actorID char(5) NOT NULL references Actors(pid),
73      directorID char(5) NOT NULL references Directors(pid),
74      movieID char(5) NOT NULL references Movies(movieID),
75   PRIMARY KEY(actorID, directorID)
76  );
77
78  -- MovieSales --
79  CREATE TABLE MovieSales (
80      salesID char(5) NOT NULL references Movies(movieID),
81      domesticSales integer,
82      foreignSales integer,
83      DVD_BluRaySales integer,
84   PRIMARY KEY(salesID)
85  );
```

3)

Address: zip --> cityOfZip, stateOfZip

People: pid --> firstName, lastName, bday, spouseFN, spouseLN, addr1, addr2, zip

Directors: pid --> directorAnniDate, directorSchool, favLensMaker

Actors: pid --> actorAnniDate, favColor

ActorPhysical: actorID --> eyeColor, hairColor, height, weight

Movies: movieID --> movieName, releaseYR, MPAA

MovieCast: actorID, directorID --> movieID

MovieSales: salesID --> domesticSales, foreignSales, DVD_BluRaySales

4)

```
87  -- Query to show all the directors with whom actor "Roger Moore" has work --
88  SELECT firstName, lastName
89  FROM People P
90  INNER JOIN Actors A ON P.pid = A.pid
91  INNER JOIN Directors D ON P.pid = D.pid
92  INNER JOIN MovieCast MC ON D.pid = MC.directorID
93  WHERE lastName = 'Moore' and firstName = 'Roger';
```