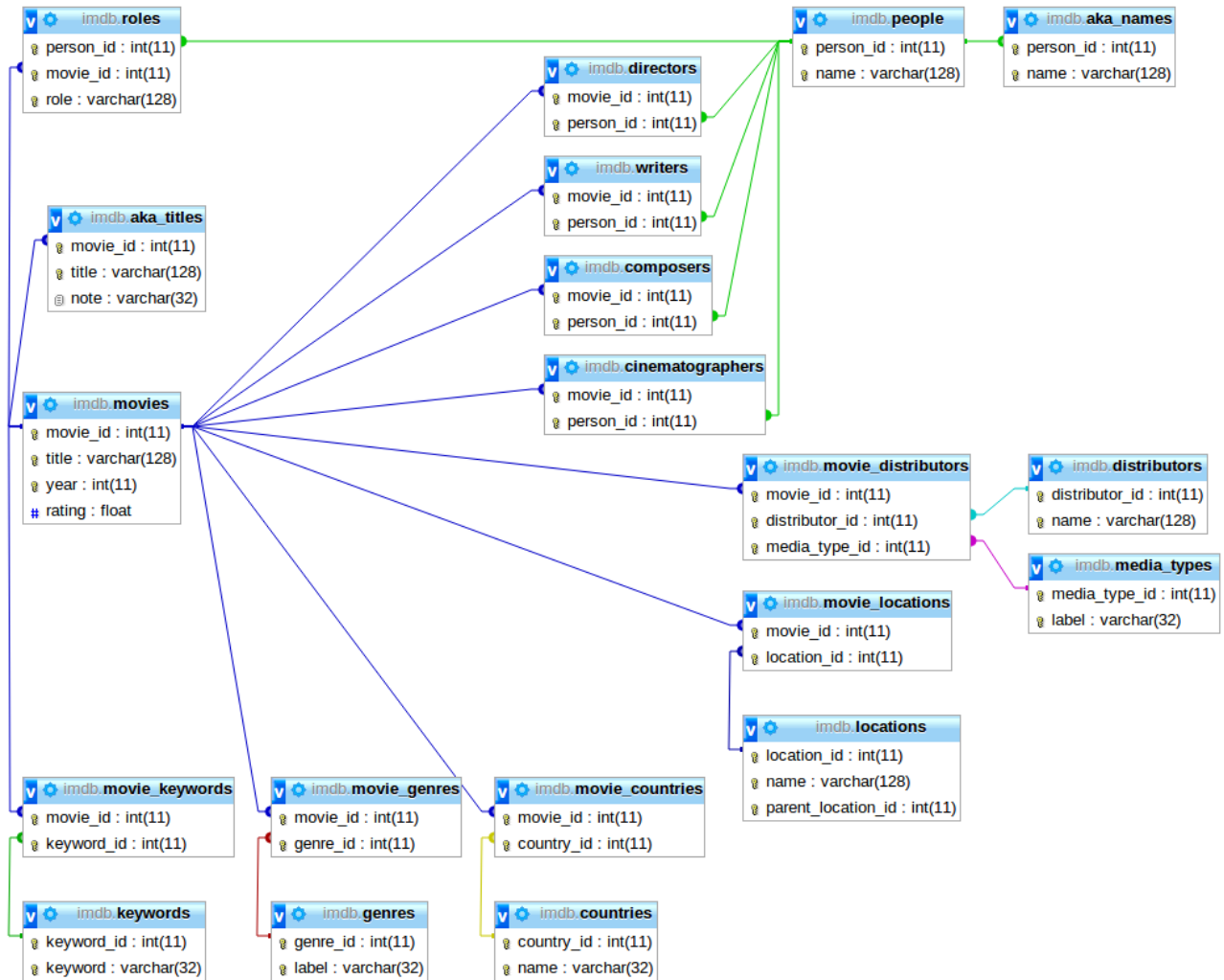


Relational Algebra: Aggregation

Database Schema

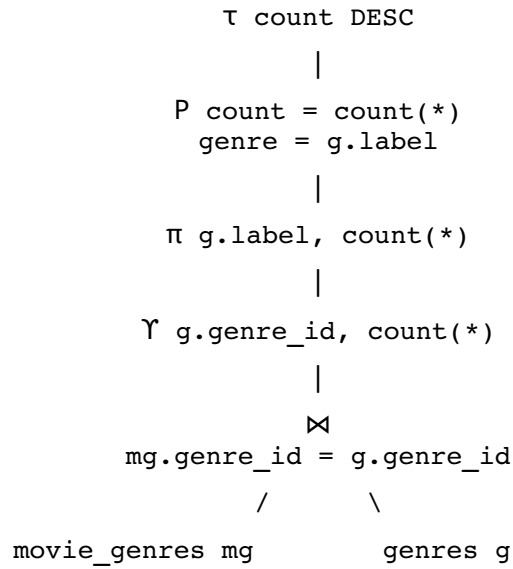


Problems

1. Find the most popular movie genres.
2. Find the people that achieved the 10 highest average ratings for the movies they cinematographed.
3. Find the actors/actresses that achieved the 10 highest average ratings for the movies they played in and have played in at least 5 movies.
4. Find the pairs of people that have directed at least 5 movies and whose careers do not overlap (i.e. The release year of a director's last movie is lower than the release year of another director's first movie).

Answers

1.



```
SELECT g.label as genre, count(*) as count
FROM genres g
JOIN movie_genres mg ON g.genre_id = mg.genre_id
GROUP BY g.genre_id
ORDER BY count DESC;
```

genre	count
Drama	154
Thriller	65
Crime	57
Adventure	54
Action	43
Mystery	41
Comedy	40
Fantasy	33
Romance	32
War	28
Sci-Fi	24
Biography	21
Family	19
Animation	17
History	15
Western	12
Film-Noir	10
Horror	9
Sport	6
Musical	5
Short	3
Music	1

2.

```

      τ average_rating DESC
      |
P average_rating = avg(m.rating)
      |
      π p.name, avg(m.rating)
      |
      γ p.person_id, avg(m.rating)
      |
      ⋈
      c.movie_id = m.movie_id
      /          \
      ⋈              movies m
p.person_id = c.person_id
      /          \
people p          cinematographers c

```

```

SELECT p.name, avg(m.rating) as average_rating
FROM people p
JOIN cinematographers c ON p.person_id = c.person_id
JOIN movies m ON c.movie_id = m.movie_id
GROUP BY p.person_id
ORDER BY average_rating DESC
LIMIT 10;

```

```

+-----+-----+
| name           | average_rating |
+-----+-----+
| Dezcallar, Jaime |           8.9 |
| Cronenweth, Jeff |           8.8 |
| Suschitzky, Peter |           8.8 |
| Lesnie, Andrew   |          8.77 |
| Burgess, Don     |           8.7 |
| Pope, Bill (I)   |           8.7 |
| Sekula, Andrzej  |          8.65 |
| Biroc, Joseph F. |           8.6 |
| Russell, John L. |           8.6 |
| Milner, Victor   |           8.6 |
+-----+-----+

```

3.

```
      τ average_rating DESC
      |
      P average_rating = avg(m.rating)
      |
      π p.name, avg(m.rating)
      |
      σ count(*) >= 5
      |
      Υ p.person_id, count(*), avg(m.rating)
      |
      ⋈
      r.movie_id = m.movie_id
      /          \
      ⋈          movies m
      p.person_id = r.person_id
      /          \
      people p      roles r
```

```
SELECT p.name, avg(m.rating) as average_rating
FROM people p
JOIN roles r ON p.person_id = r.person_id
JOIN movies m ON r.movie_id = m.movie_id
GROUP BY p.person_id
HAVING count(*) >= 5
ORDER BY average_rating DESC
LIMIT 10;
```

name	average_rating
Freeman, Morgan (I)	8.6
Guinness, Alec	8.5
Baker, Kenny (I)	8.46
Ford, Harrison (I)	8.45
Hanks, Tom	8.44
O'Connor, Frank (I)	8.42
Jackson, Samuel L.	8.42
Bale, Christian	8.42
Moorhouse, Bert	8.417
Ermey, R. Lee	8.4

4.

```

      P director_1 = p1.name
      first_movie_1 = min(m1.year)
      last_movie_1 = max(m1.year)
      director_2 = p2.name
      first_movie_2 = min(m2.year)
      last_movie_2 = max(m2.year)
      |
π p1.name, min(m1.year), max(m1.year),
  p2.name, min(m2.year) , max(m2.year)
      |
σ max(m1.year) < min(m2.year)
AND count(DISTINCT m1.movie_id) >= 5
AND count(DISTINCT m2.movie_id) >= 5
      |
Υ d1.person_id, d2.person_id,
min(m1.year), max(m1.year), min(m2.year), max(m2.year),
  count(DISTINCT m1.movie_id), count(DISTINCT m2.movie_id)
      |
      X
      /          \
      ⋈          ⋈
d1.movie_id = m1.movie_id  d2.movie_id = m2.movie_id
      /          \          /          \
      ⋈          movies m1  movies m2          ⋈
d1.person_id = p1.person_id  d2.person_id = p2.person_id
      /          \          /          \
directors d1          people p1          people p2          directors d2
```

```

SELECT
    p1.name as director_1,
    min(m1.year) as first_movie_1,
    max(m1.year) as last_movie_1,
    p2.name as director_2,
    min(m2.year) as first_movie_2,
    max(m2.year) as last_movie_2
FROM directors d1, directors d2, people p1, people p2, movies m1, movies m2
WHERE d1.person_id = p1.person_id
    AND d2.person_id = p2.person_id
    AND d1.movie_id = m1.movie_id
    AND d2.movie_id = m2.movie_id
GROUP BY d1.person_id, d2.person_id
HAVING max(m1.year) < min(m2.year)
    AND count(DISTINCT m1.movie_id) >= 5
    AND count(DISTINCT m2.movie_id) >= 5;

```

director_1	first movie 1	last movie 1	director_2	first movie 2	last movie 2
Chaplin, Charles	1921	1940	Kubrick, Stanley (I)	1956	1987
Chaplin, Charles	1921	1940	Scorsese, Martin	1976	2010
Chaplin, Charles	1921	1940	Spielberg, Steven	1975	1998
Chaplin, Charles	1921	1940	Tarantino, Quentin	1992	2009
Chaplin, Charles	1921	1940	Miyazaki, Hayao	1984	2004
Chaplin, Charles	1921	1940	Nolan, Christopher (I)	2000	2012
Chaplin, Charles	1921	1940	Wilder, Billy	1944	1960
Hitchcock, Alfred (I)	1940	1960	Scorsese, Martin	1976	2010
Hitchcock, Alfred (I)	1940	1960	Spielberg, Steven	1975	1998
Hitchcock, Alfred (I)	1940	1960	Tarantino, Quentin	1992	2009
Hitchcock, Alfred (I)	1940	1960	Miyazaki, Hayao	1984	2004
Hitchcock, Alfred (I)	1940	1960	Nolan, Christopher (I)	2000	2012
Kubrick, Stanley (I)	1956	1987	Tarantino, Quentin	1992	2009
Kubrick, Stanley (I)	1956	1987	Nolan, Christopher (I)	2000	2012
Spielberg, Steven	1975	1998	Nolan, Christopher (I)	2000	2012
Wilder, Billy	1944	1960	Scorsese, Martin	1976	2010
Wilder, Billy	1944	1960	Spielberg, Steven	1975	1998
Wilder, Billy	1944	1960	Tarantino, Quentin	1992	2009
Wilder, Billy	1944	1960	Miyazaki, Hayao	1984	2004
Wilder, Billy	1944	1960	Nolan, Christopher (I)	2000	2012