

Projetar o primeiro nome e o último nome dos atores que são diretores;

```
Algebra Relacional  Algebra multiconjunto  TRC  SQL  Editor de Grupo

select from where group having order limit

1 SELECT actors.first_name, actors.last_name
2 FROM actors
3 JOIN directors ON (actors.first_name = directors.first_name AND
4   actors.last_name = directors.last_name)
```

Projetar o primeiro nome e o último nome dos atores que não são diretores;

```
Álgebra Relacional  Álgebra multiconjunto  TRC  SQL  Editor de Grupo

select from where group having order limit

1 SELECT actors.first_name, actors.last_name
2 FROM actors
3 LEFT JOIN directors ON (actors.first_name = directors.first_name
4   AND actors.last_name = directors.last_name)
4 WHERE directors.first_name IS NULL
```

Projetar o primeiro nome e o último nome dos atores e diretores;

```
Álgebra Relacional  Álgebra multiconjunto  TRC  SQL  Editor de Grupo

select from where group having order limit

1 (SELECT actors.first_name, actors.last_name
2  FROM actors)
3 INTERSECT
4 (SELECT directors.first_name, directors.last_name
5  FROM directors)
```

Projetar o nome dos filmes que não são dirigidos por nenhum diretor;

```
Álgebra Relacional  Álgebra multiconjunto  TRC  SQL  Editor de Grupo

select from where group having order limit

1 SELECT movies.name
2 FROM movies
3 LEFT JOIN movies_directors ON movies_directors.movie_id =
4   movies.id
4 WHERE movies_directors.director_id IS NULL
```

Projetar primeiro nome e o último nome dos atores que não atuaram em pelo menos dois filmes;

```
Álgebra Relacional  Álgebra multiconjunto  TRC  SQL  Editor de Grupo
select from where group having order limit

1 SELECT actors.first_name, actors.last_name, COUNT(roles.movie_id)
  AS qnt
2 FROM actors
3 JOIN roles ON actors.id = roles.actor_id
4 GROUP BY actors.first_name, actors.last_name, actors.id
5 HAVING qnt < 2
```

Projetar, por gênero e ano, o número médio de filmes com menos de dois atores atuando.

```
Álgebra Relacional  Álgebra multiconjunto  TRC  SQL  Editor de Grupo
select from where group having order limit

1 SELECT movies_genres.genre, movies.year, COUNT(roles.actor_id) AS
  qntF, AVG(movies_genres.movie_id) AS media
2 FROM movies_genres
3 JOIN roles ON movies_genres.movie_id=roles.movie_id
4 JOIN movies ON roles.movie_id = movies.id
5 GROUP BY movies_genres.genre, movies.year
6 HAVING qntF < 2
```