

# HO04: Álgebra relacional

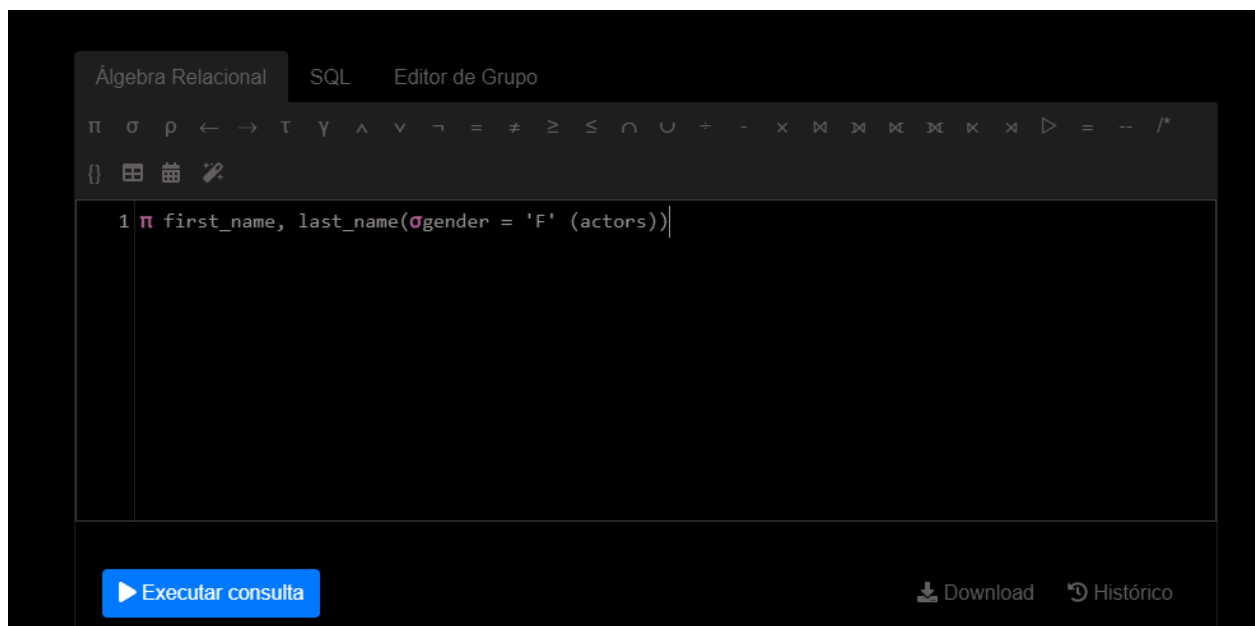
Aluno: Bernardo Ladeira Borges Kartabil

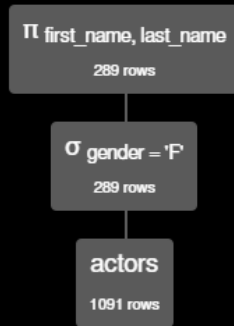
Matrícula: 838966

Turno: Manhã

## QUESTÃO 1:

$\pi$  first\_name, last\_name( $\sigma$ gender = 'F' (actors))





$\pi_{\text{first\_name, last\_name}} ( \sigma_{\text{gender} = \text{'F'}} ( \text{actors} ) )$   
Execution time: 1 ms

actors.first_name	actors.last_name
'Paula'	'Cross'
'Wendy'	'Gordon'
'Mary Elizabeth'	'Mastrantonio'
'Kimberly (I)'	'Scott'
'Emily'	'Yancy'
'Barbara'	'Coles'
'Valerie'	'Colgan'
'Holly'	'De Jong'

## QUESTÃO 2:

$$\pi \text{ name}(\sigma \text{ year} > 1999(\text{movies}))$$

Álgebra Relacional

SQL

Editor de Grupo

$\pi$

$\sigma$

$\rho$

$\leftarrow$

$\rightarrow$

$\tau$

$\gamma$

$\wedge$

$\vee$

$\neg$

$=$

$\neq$

$\geq$

$\leq$

$\cap$

$\cup$

$+$

$-$

$\times$

$\bowtie$

$\Join$

$\ltimes$

$\ltimes$

$\ltimes$

$\ltimes$

$\triangleright$

$=$

$--$

$/^*$

{ }

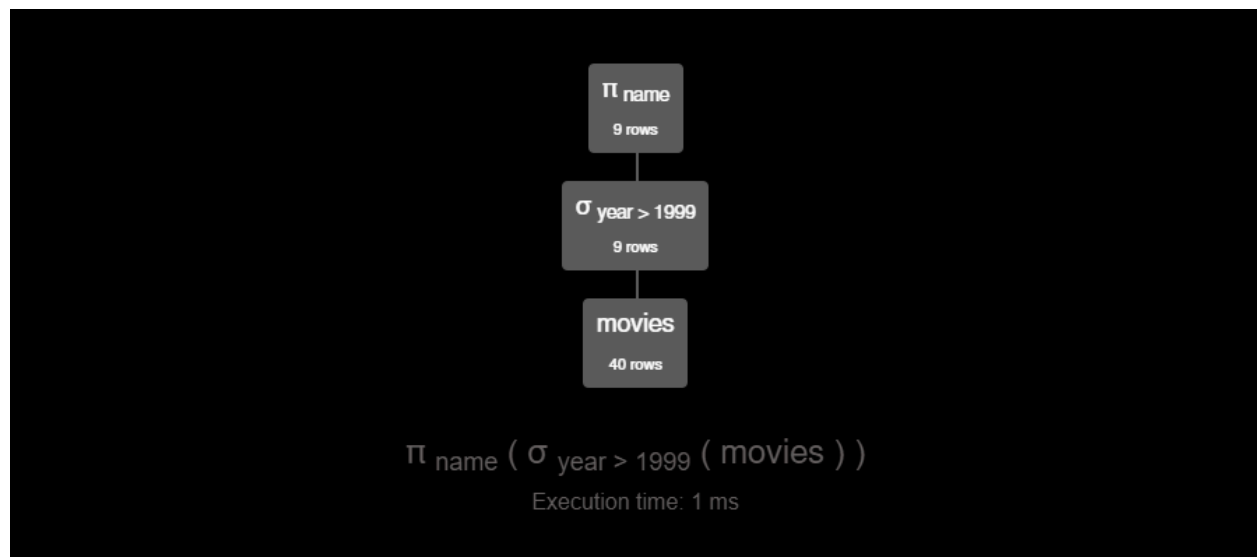
1

$\pi$  name( $\sigma$  year > 1999(movies))

Ejecutar consulta

Download

Histórico



movies.name
'Aliens of the Deep'
'Earthship TV'
'Expedition: Bismarck'
'Ghosts of the Abyss'
'Dark Angel'
'Inglorious Bastards'
'Kill Bill: Vol. 1'
'Kill Bill: Vol. 2'
'Jimmy Kimmel Live!'

### QUESTÃO 3:

-- *Junção entre movies e movies\_directors*

```
juncao_movies_directors = movies ⋈ movies.id =  
movies_directors.movie_id movies_directors
```

-- *Junção do resultado anterior com a tabela directors*

```
juncao_completa = juncao_movies_directors ⋈  
movies_directors.director_id = directors.id directors
```

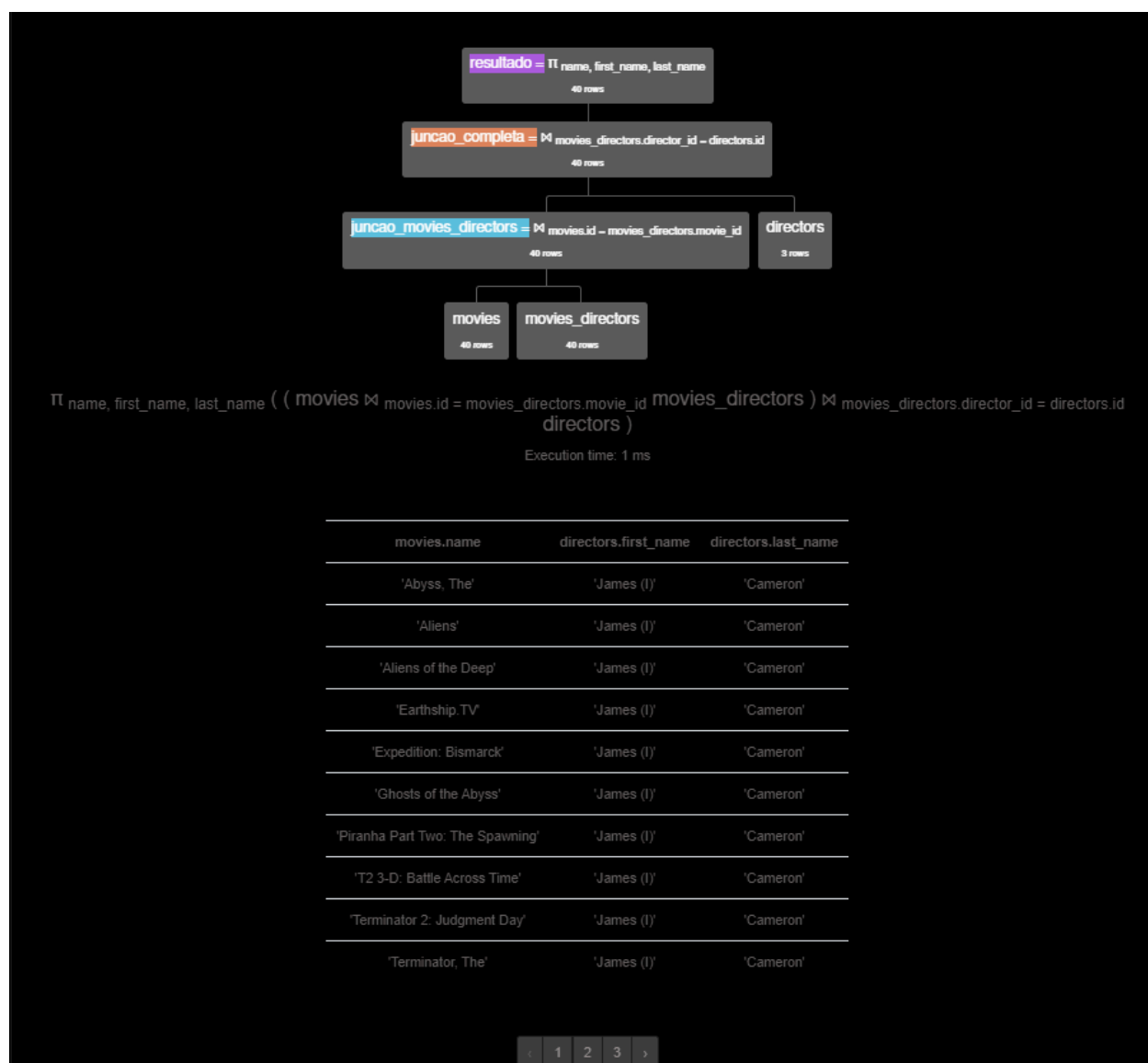
-- *Projeção para selecionar apenas as colunas desejadas*

```
resultado =  $\pi$  name, first_name, last_name (juncao_completa)
```

-- *Exibição do resultado final*

```
resultado
```





## QUESTÃO 4:

-- Filtra os filmes com ranking acima de 6

filmes\_ranking\_alto =  $\sigma$  rank > 6 (movies)

-- Junção dos filmes filtrados com a tabela de papéis (roles)

filmes\_papeis = filmes\_ranking\_alto  $\bowtie$  movies.id = roles.movie\_id roles

-- Junção com a tabela de atores para obter os nomes dos atores

juncao\_completa = filmes\_papeis  $\bowtie$  roles.actor\_id = actors.id actors

-- *Projeção das colunas desejadas*

resultado =  $\pi$  movies.name, actors.first\_name, actors.last\_name,  
roles.role (juncao\_completa)

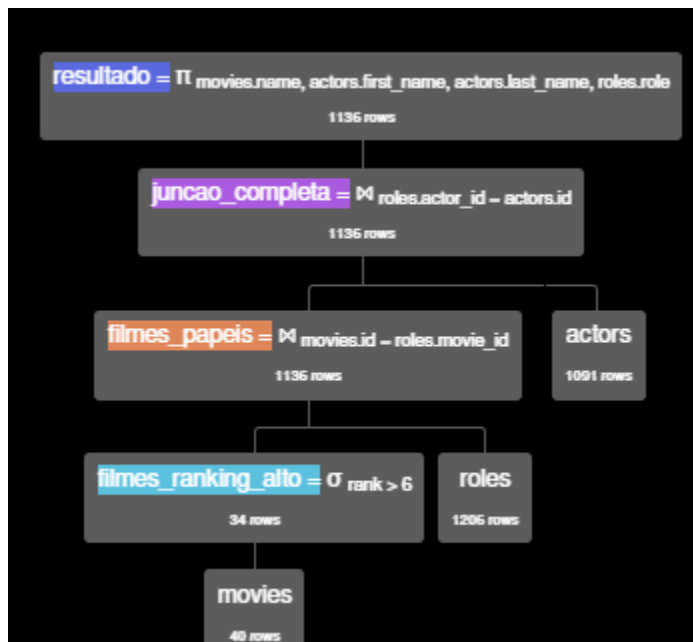
-- *Exibição do resultado final*

resultado

```
Álgebra Relacional  SQL  Editor de Grupo

1 -- Filtra os filmes com ranking acima de 6
2 filmes_ranking_alto =  $\sigma$  rank > 6 (movies)
3
4 -- Junção dos filmes filtrados com a tabela de papéis (roles)
5 filmes_papeis = filmes_ranking_alto  $\bowtie$  movies.id = roles.movie_id roles
6
7 -- Junção com a tabela de atores para obter os nomes dos atores
8 juncao_completa = filmes_papeis  $\bowtie$  roles.actor_id = actors.id actors
9
10 -- Projeção das colunas desejadas
11 resultado =  $\pi$  movies.name, actors.first_name, actors.last_name, roles.role (juncao_completa)
12
13 -- Exibição do resultado final
14 resultado
```

[▶ Executar consulta](#) [Download](#) [Histórico](#)



```
 $\pi$  movies.name, actors.first_name, actors.last_name, roles.role ( (  $\sigma$  rank > 6 ( movies )  $\bowtie$  movies.id = roles.movie_id roles )  $\bowtie$  roles.actor_id = actors.actors )
```

Execution time: 1 ms

movies.name	actors.first_name	actors.last_name	roles.role
'Abyss, The'	'Chris'	'Anastasio'	'Truck Driver'
'Abyss, The'	'Michael'	'Beach'	'Barnes'
'Abyss, The'	'John'	'Bedford Lloyd'	'Jammer Willis'
'Abyss, The'	'Michael'	'Biehn'	'Lt. Hiram Coffey'
'Abyss, The'	'Captain Kidd'	'Brewer Jr.'	'Lew Finler'
'Abyss, The'	'Leo'	'Burmester'	'Catfish De Vries'
'Abyss, The'	'Mike (I)'	'Cameron'	'Sailor with the fire extingui'
'Abyss, The'	'J. Kenneth'	'Campbell'	'DeMarco'
'Abyss, The'	'Michael (I)'	'Chapman'	'Dr. Berg'
'Abyss, The'	'Phillip'	'Darlington'	'USS Montana Crewman'

## QUESTÃO 5:

```
-- Junção entre diretores e a tabela de relação movies_directors
diretores_filmes = directors ⋈ directors.id =
movies_directors.director_id movies_directors

-- Agrupamento por diretor e contagem de filmes
resultado = γ directors.first_name, directors.last_name;
count(movies_directors.movie_id) → num_filmes (diretores_filmes)

-- Exibição do resultado final
resultado
```





Tive bastante dúvida nessa questão, mas encontrei essa explicação:

$\gamma$  atributos\_agrupamento; função\_agregação(atributo)  $\rightarrow$   
nome\_resultado (relação)

Onde:

- **atributos\_agrupamento** são os atributos pelos quais queremos agrupar (ex: genre)
- **função\_agregação(atributo)** é a função que queremos aplicar (ex: count, avg, min, max)
- $\rightarrow$  **nome\_resultado** especifica o nome da coluna de resultado
- **(relação)** é a relação sobre a qual aplicamos o agrupamento

## QUESTÃO 6:

-- Junção entre filmes e gêneros

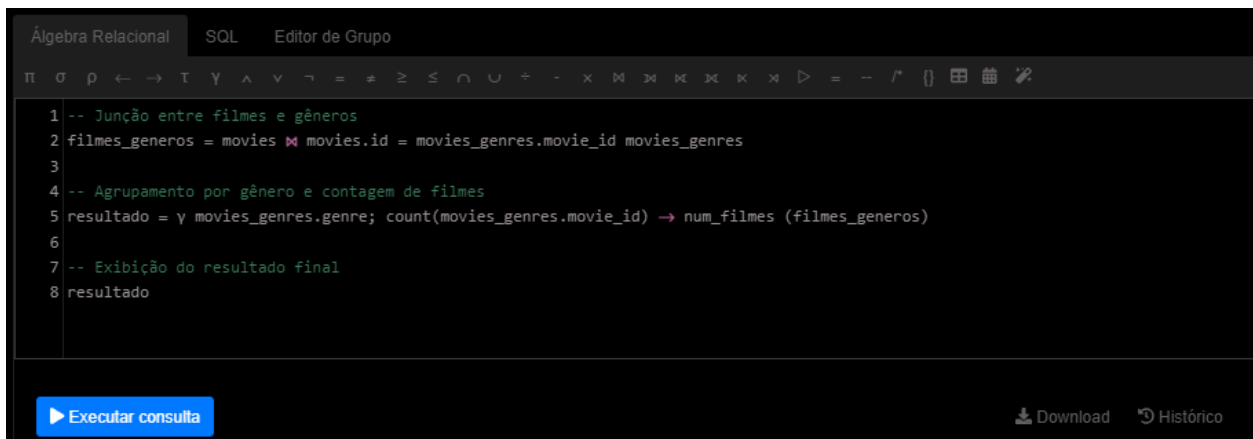
```
filmes_genres = movies ⋈ movies.id = movies_genres.movie_id  
movies_genres
```

-- Agrupamento por gênero e contagem de filmes

```
resultado =  $\gamma$  movies_genres.genre; count(movies_genres.movie_id)  $\rightarrow$   
num_filmes (filmes_genres)
```

-- Exibição do resultado final

```
resultado
```



```
1 -- Junção entre filmes e gêneros  
2 filmes_genres = movies ⋈ movies.id = movies_genres.movie_id movies_genres  
3  
4 -- Agrupamento por gênero e contagem de filmes  
5 resultado =  $\gamma$  movies_genres.genre; count(movies_genres.movie_id)  $\rightarrow$  num_filmes (filmes_genres)  
6  
7 -- Exibição do resultado final  
8 resultado
```

Executar consulta

Download Histórico



movies_genres.genre	num_filmes
'Action'	11
'Adventure'	3
'Drama'	19
'Sci-Fi'	9
'Thriller'	12
'Horror'	3
'Documentary'	6
'Family'	1
'Short'	6
'Romance'	7

1

2

Então, tentando entender minha dificuldade em relação à operação de agregação “y”, eu anotei o que fiz:

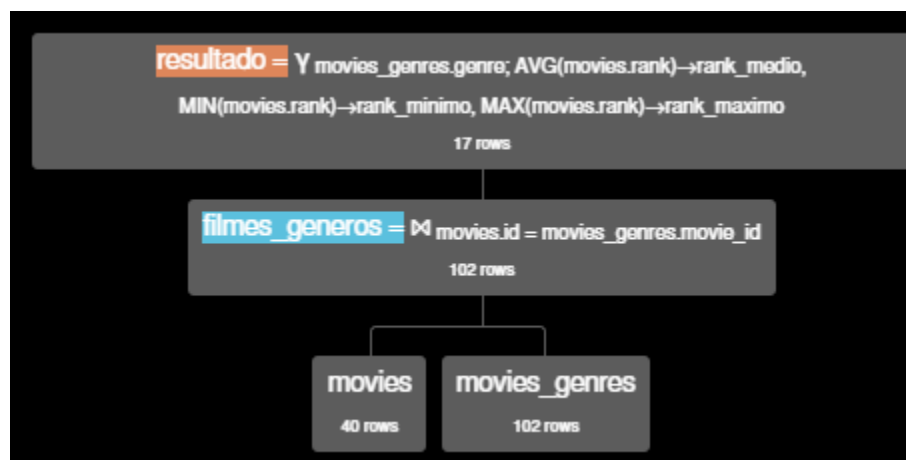
Linha:

```
resultado = γ movies_genres.genre; count(movies_genres.movie_id) →  
num_filmes (filmes_genres);
```

1. Agrupar os registros da relação **filmes\_genres** pelo atributo **movies\_genres.genre**
2. Para cada grupo, contar o número de **movie\_id** distintos
3. Nomear a coluna resultante como **num\_filmes**

## QUESTÃO 7:

```
-- Junção entre filmes e gêneros  
-- Esta linha conecta a tabela movies com a tabela movies_genres  
-- usando o ID do filme como chave de conexão  
  
filmes_genres = movies ⋈ movies.id = movies_genres.movie_id movies_genres  
  
-- Agrupamento por gênero com cálculo de estatísticas do ranking  
-- O operador γ (gamma) é usado para operações de GROUP BY e agregação  
  
resultado = γ movies_genres.genre; -- Agrupa todos os registros pelo gênero do filme  
  
avg(movies.rank) → rank_medio, -- Calcula a média dos rankings para cada gênero  
min(movies.rank) → rank_minimo, -- Encontra o menor ranking para cada gênero  
max(movies.rank) → rank_maximo -- Encontra o maior ranking para cada gênero  
(filmes_genres) -- A relação sobre a qual aplicamos o agrupamento  
  
-- Exibição do resultado final  
-- Esta linha solicita a exibição da relação resultado com todas as estatísticas  
calculadas  
resultado
```

[illegible]

movies_genres.genre	rank_medio	rank_minimo	rank_maximo
'Action'	7.927272727272728	7	8.4
'Adventure'	7.566666666666667	7	8.3
'Drama'	7.436842105263156	3.9	8.7
'Sci-Fi'	7.455555555555555	2.8	8.7
'Thriller'	7.858333333333334	7	8.4
'Horror'	6.3999999999999995	2.8	8.2
'Documentary'	6.533333333333332	5.9	7.5
'Family'	6.7	6.7	6.7
'Short'	6.083333333333333	3.9	7.4
'Romance'	6.9142857142857155	2.8	8.2

< 1 2 >

## Observação:

Professor, nessa atividade, utilizei uma inteligência artificial chamada “Claude.AI” para me auxiliar corrigindo e comentando meus códigos. Utilizei de forma consciente, pedindo para me explicar passo a passo, para eu entender melhor o conteúdo de álgebra linear! Apesar das dificuldades, sinto que aprendi!

Abraços, Bernardo.

## **FONTES EXTERNAS USADAS NA ATIVIDADE:**

Álgebra Relacional em Bancos de Dados - Operações Unárias (Seleção, Projeção e Renomeação) - <https://youtu.be/E-tlww8jDhE>

Junção de Dados - Álgebra Relacional - Junção Interna, Externa à Esquerda, à Direita e Completa - <https://youtu.be/V-CvggzYuwk>

Claude.AI - <https://claude.ai/new>