



Funções de agregação

Funções de agregação são cálculos baseados em campos da tabela que retornam uma linha de resultado. Coisas como contar, somar, mínimo e máximo são **funções de agregação**:

A função de contagem pode ser aplicada na tabela inteira "tabela", ou em campos específicos.

```
SELECT COUNT(*) FROM customers c
WHERE c.first_name IS NULL;
```

Da pra contar aplicando um filtro

Grade	123 COUNT(*)
1	0

A função de soma precisa ser feita em campos numéricos

```
SELECT SUM(number_seats) FROM rooms;
```

Grade	123 SUM(number_seats)
1	144

As funções MIN/MAX trazem o valor mais baixo e o mais alto de uma coluna

```
SELECT MAX(min_length) FROM films;
SELECT MIN(min_length) FROM films;
```

Grade	123 MAX(min_length)
1	153

Grade	123 MIN(min_length)
1	88

A função de média é o cálculo básico, soma tudo e divide pela quantidade precisa ser feita em campos numéricos

```
SELECT AVG(min_length) FROM films;
```

Grade	123 AVG(min_length)
1	115,0833

As **funções de agregação** são bem mais úteis quando combinadas com **agrupadores**. A ideia de agrupar as funções é pra ter um **panorama mais informativo sobre os dados que estão sendo calculados**:

Contar a quantidade de IDs de clientes que aparecem na tabela de reservas é uma maneira de descobrir quantas reservas um cliente fez

Só que eu preciso saber a ocorrência das reservas por CADA CLIENTE, então a contagem precisa ser agrupada

```
SELECT customer_id, COUNT(id) FROM bookings
GROUP BY customer_id;
```

Grade	123 customer_id	123 COUNT(id)
1	1	16
2	2	4
3	3	6
4	4	13
5	5	5
6	6	5
7	7	9

Cliente 1 fez 16 reservas

Conta as reservas desses filmes

E traz o nome e a data de início da exibição do filme, e o nome dos clientes que fizeram a reserva

```
SELECT f.name, s.start_time, c.first_name, c.last_name, COUNT(b.id) FROM films f
JOIN screenings s ON f.id = s.film_id
JOIN bookings b ON s.id = b.screening_id
JOIN customers c ON b.customer_id = c.id
GROUP BY f.name, c.first_name, c.last_name, s.start_time
ORDER BY f.name;
```

Grade	123 name	123 start_time	123 first_name	123 last_name	123 COUNT(b.id)
1	Blade Runner	2017-10-22 16:30:00	Benjamin	Rabbit	1
2	Blade Runner	2017-10-22 13:30:00	Cherry	Wang	1
3	Blade Runner	2017-10-15 14:00:00	David	Paul	1
4	Blade Runner	2017-10-22 19:30:00	David	Paul	1
5	Blade Runner	2017-10-22 16:30:00	Josh	Reynolds	1
6	Blade Runner	2017-10-22 19:30:00	Preston	Peters	1
7	Blade Runner	2017-10-15 17:00:00	Ron	Potter	1

Se você conta então precisa agrupar o resultado

O WHERE de um GROUP BY é o HAVING.



SubQueries

SubQueries são queries do tipo **SELECT** dentro de **outras queries**, elas podem ser **relacionadas** ou **não relacionadas** com a **outerQuery** (a query de fora). Então a ideia principal da subquerie é alimentar a query maior:

A outerQuery usa esse resultado para performar

```
SELECT id, start_time FROM screenings s
WHERE s.film_id IN
(SELECT id FROM films WHERE min_length > 120)
```

A subconsulta/innerQuery retorna um set de resultado. EX: 1,2,3,4

