



Aula #4: Agrupamentos

☰ Módulo	SQL Básico
# Aula	10
🕒 Created	@May 8, 2022 6:57 PM
☑ Reviewed	☑
📎 Material PDF	

Aula #4: Agrupamentos

Objetivo da Aula:

- ☐ O que são agrupamento?
- ☐ Agrupamento em SQL com uma ou mais dimensões.
- ☐ Exercícios

Conteúdo:

▼ Operações Matemáticas em Agrupamentos

As operações matemáticas resumem um conjunto de dados em apenas um número.
Podem ser realizadas dentro de um agrupamento.

▼ Exemplo:

▼ Agrupamentos no SQL

Os agrupamento em SQL funcionam exatamente como as tabelas dinâmicas do Excel ou Google Sheets. Uma ou mais operações são aplicadas um ou mais grupos e subgrupos.

▼ Exemplos:

▼ Agrupamento com 1 variável (dimensão)

```
SELECT
  p.product_category_name,
  COUNT( p.product_id ),
  SUM( p.product_weight_g ),
  AVG( p.product_weight_g ),
  MAX( p.product_height_cm ),
  MIN( p.product_width_cm ),
  COUNT( DISTINCT p.product_length_cm )
FROM products p
GROUP BY p.product_category_name
```

```
SELECT
  oi.seller_id ,
  COUNT( DISTINCT oi.order_id ) AS pedidos,
  COUNT( oi.product_id ) AS items,
  AVG( oi.price ) AS preco_medio
FROM order_items oi
GROUP BY oi.seller_id
```

▼ Agrupamento com 2 variáveis (dimensões)

```
SELECT
  s.seller_state ,
  s.seller_city,
  COUNT( s.seller_id ),
  SUM( s.seller_zip_code_prefix )
FROM sellers s
GROUP BY s.seller_state , s.seller_city
```

```
SELECT
    DATE( oi.shipping_limit_date ) AS data_ ,
    COUNT( oi.seller_id ) AS vendedores,
    COUNT( DISTINCT oi.order_id ) AS pedidos,
    COUNT( oi.product_id ) AS items,
    AVG( oi.price ) AS preco_medio
FROM order_items oi
GROUP BY DATE( oi.shipping_limit_date )
```

▼ Resumo:

1. As operações matemáticas sobre agrupamentos funcionam da mesma forma que as tabelas dinâmicas no Excel ou Google Sheets.
2. Em SQL, usamos o **GROUP BY** para indicar a coluna (dimensão) do agrupamento.
3. Podemos realizar agrupamentos a partir de uma ou mais colunas. Todas as colunas do agrupamento precisam ser indicadas no **GROUP BY**.

▼ Exercícios do PA Bond

▼ As perguntas do CEO:

▼ 1. Qual o número de clientes únicos de todos os estados?

```
SELECT
    c.customer_id ,
    COUNT( DISTINCT c.customer_id ) AS numero_clientes
FROM customer c
GROUP BY c.customer_state
```

▼ 2. Qual o número de cidades únicas de todos os estados?

```
SELECT
    customer_state ,
    COUNT( DISTINCT customer_city ) AS numero_cidades
FROM customer c
GROUP BY customer_state
```

▼ 3. Qual o número de clientes únicos por estado e por cidade?

```
SELECT
    c.customer_state,
    c.customer_city ,
    COUNT( DISTINCT c.customer_id ) AS clientes
FROM customer c
GROUP BY c.customer_state , c.customer_city
```

▼ 4. Qual o número de clientes únicos por cidade e por estado?

```
SELECT
    c.customer_city ,
    c.customer_state,
    COUNT( DISTINCT c.customer_id ) AS clientes
FROM customer c
GROUP BY c.customer_city, c.customer_state
```

▼ 5. Qual o número total de pedidos únicos acima de R\$ 3.500 por cada vendedor?

```
SELECT
    seller_id,
    COUNT( DISTINCT order_id )
FROM order_items oi
WHERE price > 3500
GROUP BY seller_id
```

▼ 6. Qual o número total de pedidos únicos, a data mínima e máxima de envio, o valor máximo, mínimo e médio do frete dos pedidos acima de R\$ 1.100 por cada vendedor?

```
SELECT
    seller_id,
    COUNT( DISTINCT order_id ) AS pedidos_unicos,
    MIN( shipping_limit_date ) AS data_minima_envio,
```

```

MAX( shipping_limit_date ) AS data_maxima_envio,
AVG( freight_value )      AS valor_medio_frete_medio,
MIN( freight_value)       AS valor_minimo_frete,
MAX( freight_value)       AS valor_maximo_frete
FROM order_items oi
WHERE price > 1100
GROUP BY seller_id

```

▼ **7. Qual o valor médio, máximo e mínimo do preço de todos os pedidos de cada produto?**

```

SELECT
  oi.product_id,
  AVG( oi.price ) AS preco_medio,
  MIN( oi.price ) AS preco_minimo,
  MAX( oi.price ) AS preco_maximo
FROM order_items oi
GROUP BY oi.product_id

```

▼ **8. Qual a quantidade de vendedores distintos que receberam algum pedido antes do dia 23 de setembro de 2016 e qual foi o preço médio desses pedidos?**

```

SELECT
  oi.shipping_limit_date,
  COUNT( seller_id ) AS vendedores,
  AVG( oi.price )    AS preco_medio
FROM order_items oi
WHERE shipping_limit_date < '2016-09-23 00:00:00'
GROUP by oi.shipping_limit_date

```

▼ **10. Qual a quantidade de pedidos por tipo de pagamentos?**

```

SELECT
  payment_type,
  COUNT( op.order_id ) as pedidos
FROM order_payments op
GROUP BY op.payment_type

```

▼ 11. Qual a quantidade de pedidos, a média do valor do pagamento e o número máximo de parcelas por tipo de pagamentos?

```
SELECT
    payment_type,
    COUNT( op.order_id )           AS pedidos,
    AVG( op.payment_value )       AS pagamento_medio,
    MAX( op.payment_installments ) AS maior_numero_parcelas
FROM order_payments op
GROUP BY op.payment_type
```

▼ 12. Qual a valor mínimo, máximo, médio e as soma total paga por cada tipo de pagamento e número de parcelas disponíveis?

```
SELECT
    payment_type,
    payment_installments,
    MIN( payment_value ) AS pagamento_minimo,
    MAX( payment_value ) AS pagamento_maximo,
    AVG( payment_value ) AS pagamento_medio,
    SUM( payment_value ) AS pagamento_total
FROM order_payments op
GROUP BY payment_type, payment_installments
```

▼ 14. Qual a média de pedidos por cliente?

```
SELECT
    customer_id ,
    AVG( order_id ) AS media_pedidos
FROM orders o
GROUP BY customer_id
```

▼ 15. Qual a quantidade de pedidos realizados por cada status do pedido, a partir do dia 23 de Setembro de 2016?

```
SELECT
    order_status ,
    COUNT( order_id ) AS pedidos
```

```
FROM orders o
WHERE order_approved_at > '2016-09-23 00:00:00'
GROUP BY order_status
```

▼ **16. Qual a quantidade de pedidos realizados por dia, a partir do dia 23 de Setembro de 2016?**

```
SELECT
    DATE( order_approved_at ) AS data_ ,
    COUNT( order_id )          AS pedidos
FROM orders o
WHERE order_approved_at > '2016-09-23 00:00:00'
GROUP BY DATE( order_approved_at )
```

▼ **17. Quantos produtos estão cadastrados na empresa por categoria?**

```
SELECT
    product_category_name ,
    COUNT( DISTINCT product_id ) AS produtos
FROM products p
GROUP BY product_category_name
```

▼ **Na próxima aula ...**

Aula #5: Operadores de comparação