INSTRUÇÃO GROUP BY

Modelagem de Dados

INTRODUÇÃO

Vamos aprender a trabalhar com as cláusulas GROUP BY e HAVING em SQL.

Elas são muito úteis quando queremos agrupar informações e aplicar funções de agregação, como somar, contar ou calcular a média.

Além disso, o **HAVING** nos permite filtrar os resultados após a agregação, similar ao **WHERE**, mas aplicável em conjuntos agrupados.

RELEMBRANDO NOSSO BANCO DE DADOS

Vamos usar a base de dados **DB_EMPRESA** que já conhecemos, e como sabemos essa base contém três tabelas principais:

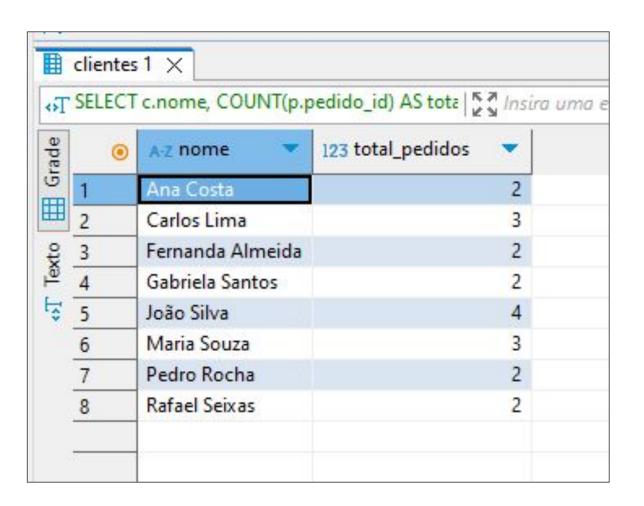
- Clientes: Contém informações sobre os clientes.
- Pedidos: Armazena os pedidos feitos pelos clientes.
- ItensPedido: Registra os itens de cada pedido.

Você já conhece a estrutura dessas tabelas com exemplos reais de dados, então vamos focar diretamente em como o **GROUP BY** e o **HAVING** funcionam.

GROUP BY: Agrupando Resultados

Imagine que queremos saber quantos pedidos cada cliente fez. Podemos usar o **GROUP BY** para agrupar os pedidos por cliente e contar quantos pedidos há para cada um. Vamos fazer isso!

```
SELECT
    c.nome, COUNT(p.pedido_id) AS total_pedidos
FROM
    Clientes c
    INNER JOIN Pedidos p ON c.cliente_id = p.cliente_id
GROUP BY
    c.nome;
```



GROUP BY: Agrupando Resultados

Explicação:

Estamos selecionando o nome do cliente da tabela **Clientes** e contando quantos **pedido_id** existem na tabela **Pedidos**.

A cláusula GROUP BY agrupa os resultados pelo nome do cliente.

Assim, para cada cliente, teremos o número total de pedidos feitos.

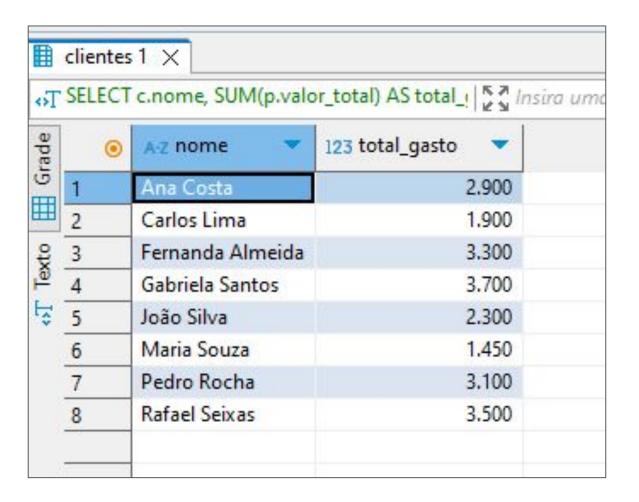
INSTRUÇÃO DE AGREGAÇÃO

FUNÇÕES DE AGREGAÇÃO

Além de contar, também podemos somar, calcular médias e muito mais.

Por exemplo, se quisermos saber o valor total dos pedidos por cliente, podemos usar o **SUM()**:

```
SELECT
   c.nome, SUM(p.valor_total) AS total_gasto
FROM
   Clientes c
   INNER JOIN Pedidos p ON c.cliente_id = p.cliente_id
GROUP BY
   c.nome;
```



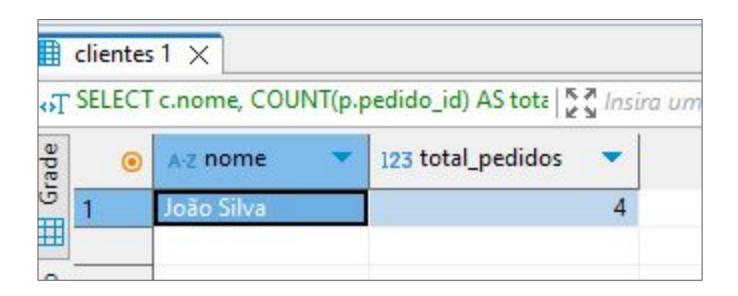
HAVING: Filtrando após o agrupamento

O HAVING é usado quando queremos filtrar resultados após o GROUP BY.

Imagine que queremos ver apenas os clientes que fizeram mais de 3 pedidos.

O WHERE não funcionaria nesse caso porque estamos lidando com um valor agregado, então usamos HAVING:

```
SELECT
    c.nome, COUNT(p.pedido_id) AS total_pedidos
FROM
    Clientes c
    INNER JOIN Pedidos p ON c.cliente_id = p.cliente_id
GROUP BY
    c.nome
HAVING COUNT(p.pedido_id) > 3;
```



HAVING: Filtrando após o agrupamento

Explicação:

A função **COUNT()** nos dá o número total de pedidos por cliente.

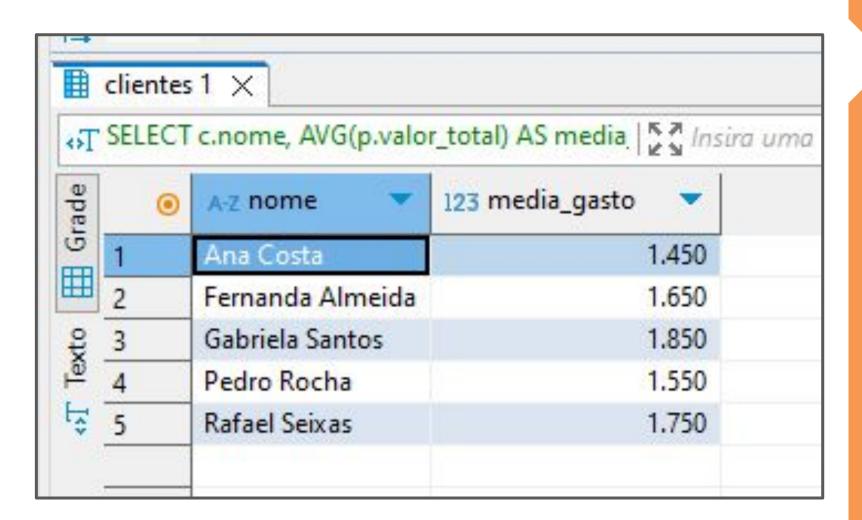
HAVING COUNT(p.pedido_id) > 3 vai exibir apenas os clientes que têm mais de 3 pedidos.

EXEMPLO COMBINADO

Pedido Médio por Cliente

Vamos calcular o **valor médio de pedidos <u>por cliente</u>**, mas queremos exibir apenas os clientes que gastaram **mais de R\$ 1000 em média**.

```
SELECT
  c.nome, AVG(p.valor_total) AS media_gasto
FROM
   Clientes c
   INNER JOIN Pedidos p ON c.cliente_id = p.cliente_id
GROUP BY
   c.nome
HAVING
  AVG(p.valor_total) > 1000;
```



EXERCÍCIOS

Lista de Exercícios - 02

- Disponível no AVA Unidade 1 Seção 1 Laboratório
- Entregar escrito à mão até antes da prova
- Vale **500 pontos**