

Evidências Banco de Dados: "db_revenda_rebeca"

- Seleciona todos os clientes onde o nome começa com a letra A

```
select * from clientes c where nome like 'A%';
```

clientes 1 X

`select * from clientes c where nome like 'A'` | *Insira uma expressão SQL para filtrar os resultados (use Ctrl+Espaço)*

Grade	123 id	A-Z nome	A-Z email	A-Z telefone	data_cadastro
1	1	Ana Souza	ana@email.com	119999999	2025-08-27 08:25:53.990

- Explica a consulta anterior

```
explain select * from clientes where nome like 'A%';
```

Resultados 1 X

`explain select * from clientes where nome` | *Insira uma expressão SQL*

Grade	A-Z QUERY PLAN
1	Seq Scan on clientes (cost=0.00..1.12 rows=1 width=506)
2	Filter: ((nome)::text ~~ 'A% '::text)

Texto

```
create index idx_clientes_nome on clientes(nome);
```

- Explica para que serve a tabela clientes

```
explain select * from clientes where nome like 'E%';
```

Resultados 1 X

`explain select * from clientes where nome` | *Insira uma expressão SQL*

Grade	A-Z QUERY PLAN
1	Seq Scan on clientes (cost=0.00..1.12 rows=1 width=480)
2	Filter: ((nome)::text ~~ 'E% '::text)


to

- Mudar a coluna telefone de clientes de varchar para int, porém deu erro

```
alter table clientes alter column telefone set data type int;
```

Resultados 1 X

`explain select * from clientes where nome` | *Insira uma expressão SQL para*


 Erro SQL [42804]: ERRO: coluna "telefone" não pode ser convertida automaticamente para tipo integer
Dica: Talvez seja necessário especificar "USING telefone::integer".

- Mudar a coluna id da tabela clientes para varchar, porém deu erro

```
alter table clientes alter column id set data type varchar(20);
```

Resultados 1 X

explain select * from clientes where nome | Insira uma expressão SQL para filtrar

 Erro SQL [0A000]: ERRO: não é possível alterar o tipo de dados de uma coluna usada por uma visão ou regra
Detalhe: regra_RETURN em visão vw_vendas_clientes depende da coluna "id"

- Criação dos usuários e permissões para eles

```
create user rebeca with password 'senha123';
grant all privileges on all tables in schema public to rebeca;
create user lola with password 'senha456';
grant select on clientes to lola;
```

- Atualizar a coluna email da tabela cliente

```
update clientes set email = null where id < 5;
```

Estatísticas 1 X

Name	Value
Updated Rows	4
Query	update clientes set email = null where id < 5
Start time	Wed Aug 27 09:04:37 BRT 2025
Finish time	Wed Aug 27 09:04:37 BRT 2025

- Junta o nome da tabela cliente e a rua da tabela endereço e retorna o cliente e a rua do cliente

```
select c.nome, e.rua from clientes c inner join enderecos e on c.id = e.cliente_id;
```

clientes(+) 1 X

select c.nome, e.rua from clientes c inner

Grade	A-Z nome	A-Z rua
1	Ana Souza	Rua A
2	Carlos Silva	Av Atlântica
3	Fernanda Lima	Rua B
4	João Pedro	Rua C
5	Mariana Costa	Rua D
6	Rodrigo Alves	Rua E
7	Patrícia Gomes	Rua F
8	Gabriel Martins	Rua G
9	Juliana Rocha	Rua H
10	Ricardo Dias	Rua I

- Junta a tabela cliente com a tabela venda, o inner Join garante que somente os clientes que tem vendas registrados sejam retornados.

```
select c.nome, v.id as venda_id, v.valor_total from clientes c inner join vendas v on c.id = v.cliente_id;
```

clientes(+) 1 X

select c.nome, v.id as venda_id, v.valor_to | Insira uma expressão SQL

Ctrl+clique para abrir o console SQL

Grade	nome	123	
1	Ana Souza	1	1.470
2	Carlos Silva	2	850
3	Fernanda Lima	3	900
4	João Pedro	4	530
5	Mariana Costa	5	1.200
6	Rodrigo Alves	6	700
7	Patrícia Gomes	7	1.500
8	Gabriel Martins	8	600
9	Juliana Rocha	9	450
10	Ricardo Dias	10	780

- Mostra quais produtos estão em cada venda e a quantidade e faz só aparecer vendas que têm produtos cadastrados.

```
select v.id as venda_id, p.nome as produto, vp.quantidade from vendas v inner join venda_produto vp on v.id = vp.venda_id inner join produtos p on vp.produto_id = p.id;
```

vendas(+) 1 X

select v.id as venda_id, p.nome as produto | Insira uma expressão SQL para filtrar os resu

Grade	123 venda_id	A-Z produto	123 quantidade
1	1	Assistente do Vilão	1
2	1	O Amor Não é Óbvio	1
3	2	Assistente do Vilão	1
4	3	A Hipótese do Amor	1
5	4	O Despertar da Lua Caída	1
6	5	O Primeiro a Morrer no Final	2
7	6	O Mundo de Sofia	1
8	7	Assassinato no Expresso Oriente	2
9	8	Saboroso Cadaver	1
10	9	O Mundo de Sofia	1

- Lista produtos com seu fornecedor e preço.

```
select p.nome, f.nome as fornecedor, p.preco from produtos p inner join fornecedores f on p.fornecedor_id = f.id;
```

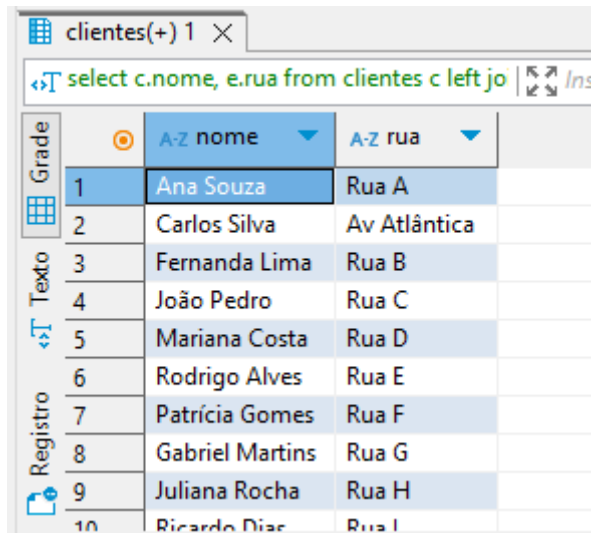
produtos(+) 1 X

select p.nome, f.nome as fornecedor, p.pr | Insira uma expressão SQL para filtrar os re

Grade	A-Z nome	A-Z fornecedor	123 preco
1	Assistente do Vilão	Editora Globo	50
2	O Amor Não é Óbvio	Editora Galera	62
3	Um Estudo em Vermelho	Editora Pé da Letra	70
4	O Primeiro a Morrer no Final	Editora Intrínseca	70
5	O Mundo de Sofia	Editora Seguinte	50
6	O Despertar da Lua Caída	Editora Harlequin	53
7	Assassinato no Expresso Oriente	Editora L&PM pocket	78
8	Saboroso Cadaver	Editora Dark Side	60
9	A Hora da Estrela	Editora Rocco	67
10	A Hipótese do Amor	Editora Arqueiro	90

- Junta o nome da tabela cliente e a rua da tabela endereço e retorna o cliente e a rua do cliente

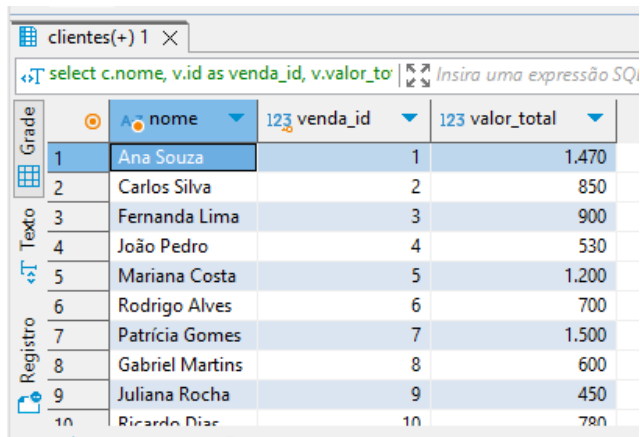
```
select c.nome, e.rua from clientes c left join endereços e on c.id = e.cliente_id;
```



	A-Z nome	A-Z rua
1	Ana Souza	Rua A
2	Carlos Silva	Av Atlântica
3	Fernanda Lima	Rua B
4	João Pedro	Rua C
5	Mariana Costa	Rua D
6	Rodrigo Alves	Rua E
7	Patrícia Gomes	Rua F
8	Gabriel Martins	Rua G
9	Juliana Rocha	Rua H
10	Ricardo Dias	Rua I

- Junta o nome do cliente, com o id da venda, o valor total da compra

```
select c.nome, v.id as venda_id, v.valor_total from clientes c left join vendas v on c.id = v.cliente_id;
```



	A-Z nome	123 venda_id	123 valor_total
1	Ana Souza	1	1.470
2	Carlos Silva	2	850
3	Fernanda Lima	3	900
4	João Pedro	4	530
5	Mariana Costa	5	1.200
6	Rodrigo Alves	6	700
7	Patrícia Gomes	7	1.500
8	Gabriel Martins	8	600
9	Juliana Rocha	9	450
10	Ricardo Dias	10	790

- Junta o nome do cliente, com o id da venda, o valor total da compra

```
select c.nome, v.id as venda_id, v.valor_total from clientes c right join vendas v on c.id = v.cliente_id;
```

clientes(+) 1 X			
select c.nome, v.id as venda_id, v.valor_to Insira uma expressão SQL			
	nome	venda_id	valor_total
1	Ana Souza	1	1.470
2	Carlos Silva	2	850
3	Fernanda Lima	3	900
4	João Pedro	4	530
5	Mariana Costa	5	1.200
6	Rodrigo Alves	6	700
7	Patrícia Gomes	7	1.500
8	Gabriel Martins	8	600
9	Juliana Rocha	9	450
10	Ricardo Dias	10	780

- Junta o id da venda, o produto e a quantidade

```
select v.id as venda_id, p.nome as produto, vp.quantidade from vendas v left join venda_produto vp on v.id = vp.venda_id left join produtos p on vp.produto_id = p.id;
```

vendas(+) 1 X			
select v.id as venda_id, p.nome as produto Insira uma expressão SQL para filtrar os			
	venda_id	produto	quantidade
1	1	Assistente do Vilão	1
2	1	O Amor Não é Óbvio	1
3	2	Assistente do Vilão	1
4	3	A Hipótese do Amor	1
5	4	O Despertar da Lua Caída	1
6	5	O Primeiro a Morrer no Final	2
7	6	O Mundo de Sofia	1
8	7	Assassinato no Expresso Oriente	2
9	8	Saboroso Cadaver	1
10	9	O Mundo de Sofia	1

- Junta o id da venda, o produto e a quantidade

```
select v.id as venda_id, p.nome as produto, vp.quantidade from vendas v right join venda_produto vp on v.id = vp.venda_id right join produtos p on vp.produto_id = p.id;
```

vendas(+) 1 X			
select v.id as venda_id, p.nome as produto Insira uma expressão SQL para filtrar os			
	venda_id	produto	quantidade
1	1	Assistente do Vilão	1
2	1	O Amor Não é Óbvio	1
3	2	Assistente do Vilão	1
4	3	A Hipótese do Amor	1
5	4	O Despertar da Lua Caída	1
6	5	O Primeiro a Morrer no Final	2
7	6	O Mundo de Sofia	1
8	7	Assassinato no Expresso Oriente	2
9	8	Saboroso Cadaver	1
10	9	O Mundo de Sofia	1

- Junta o nome do produto, com o fornecedor e com o preço

```
select p.nome, f.nome as fornecedor, p.preco from produtos p left join fornecedores f on p.fornecedor_id = f.id;
```

The screenshot shows a database query interface. At the top, the query is: `select p.nome, f.nome as fornecedor, p.preco from produtos p left join fornecedores f on p.fornecedor_id = f.id;`. Below the query, there's a table with 4 columns: 'nome', 'fornecedor', and 'preco'. The table has 10 rows of data. The interface also shows a sidebar with icons for 'Grade', 'Texto', and 'Registro'.

	nome	fornecedor	preco
1	Assistente do Vilão	Editora Globo	50
2	O Amor Não é Óbvio	Editora Galera	62
3	Um Estudo em Vermelho	Editora Pé da Letra	70
4	O Primeiro a Morrer no Final	Editora Intrínseca	70
5	O Mundo de Sofia	Editora Seguinte	50
6	O Despertar da Lua Caída	Editora Harlequin	53
7	Assassinato no Expresso Oriente	Editora L&PM pocket	78
8	Saboroso Cadaver	Editora Dark Side	60
9	A Hora da Estrela	Editora Rocco	67
10	A História do Amor	Editora Arqueiro	90

- Junta o nome do produto, com o fornecedor e com o preço

```
select p.nome, f.nome as fornecedor, p.preco from produtos p right join fornecedores f on p.fornecedor_id = f.id;
```

The screenshot shows a database query interface. At the top, the query is: `select p.nome, f.nome as fornecedor, p.preco from produtos p right join fornecedores f on p.fornecedor_id = f.id;`. Below the query, there's a table with 4 columns: 'nome', 'fornecedor', and 'preco'. The table has 10 rows of data. The interface also shows a sidebar with icons for 'Grade', 'Texto', and 'Registro'.

	nome	fornecedor	preco
1	Assistente do Vilão	Editora Globo	50
2	O Amor Não é Óbvio	Editora Galera	62
3	Um Estudo em Vermelho	Editora Pé da Letra	70
4	O Primeiro a Morrer no Final	Editora Intrínseca	70
5	O Mundo de Sofia	Editora Seguinte	50
6	O Despertar da Lua Caída	Editora Harlequin	53
7	Assassinato no Expresso Oriente	Editora L&PM pocket	78
8	Saboroso Cadaver	Editora Dark Side	60
9	A Hora da Estrela	Editora Rocco	67
10	A História do Amor	Editora Arqueiro	90

Diferenças entre os “joins”

RIGHT JOIN = garante que **todos os fornecedores apareçam**, mesmo sem produtos.

INNER JOIN só mostraria fornecedores que tivessem pelo menos 1 produto.

LEFT JOIN seria usado se quiséssemos garantir todos os **produtos**, mesmo sem fornecedor (menos comum aqui).