

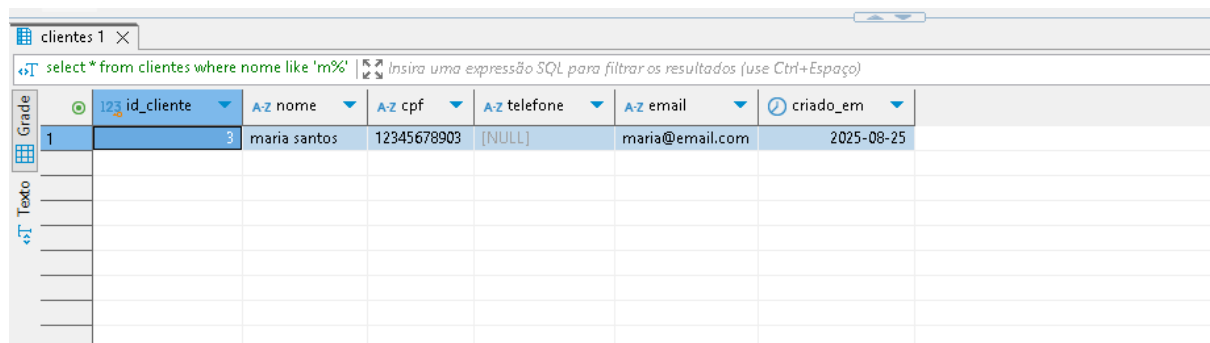
Documento de Evidências – Banco de Dados Revenda

Aluno: Maria Eduarda de Paula Tamancoldi

1- Consulta filtrando clientes com LIKE

Código utilizado:

```
select * from clientes where nome like 'm%';
```



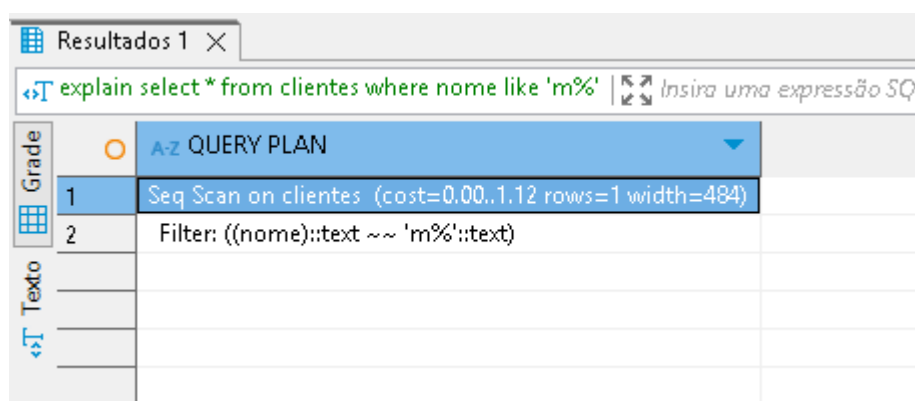
	id_cliente	nome	cpf	telefone	email	criado_em
1	3	maria santos	12345678903	[NULL]	maria@email.com	2025-08-25

Clientes filtrados com LIKE antes da criação do índice.

2- EXPLAIN antes do índice

Código utilizado:

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



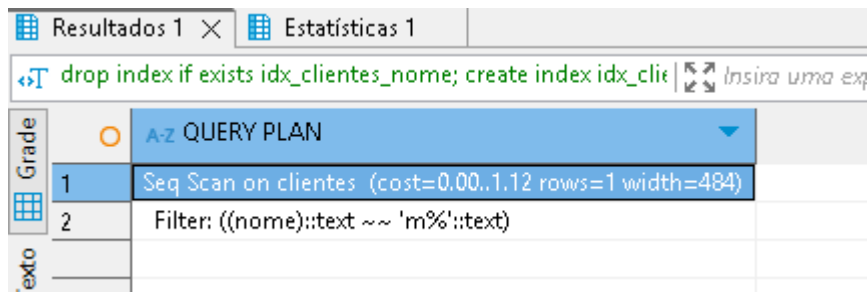
	A-Z QUERY PLAN
1	Seq Scan on clientes (cost=0.00..1.12 rows=1 width=484)
2	Filter: ((nome)::text ~~ 'm% '::text)

EXPLAIN antes do índice, mostrando Seq Scan.

3- Criação do índice e EXPLAIN depois

Código utilizado:

```
drop index if exists idx_clientes_nome;  
create index idx_clientes_nome on clientes(nome);  
explain select * from clientes where nome like 'm%';
```



The screenshot shows a database interface with a query plan for the SQL statement: `drop index if exists idx_clientes_nome; create index idx_clientes_nome on clientes(nome); explain select * from clientes where nome like 'm%';`. The query plan is displayed in a table with the following structure:

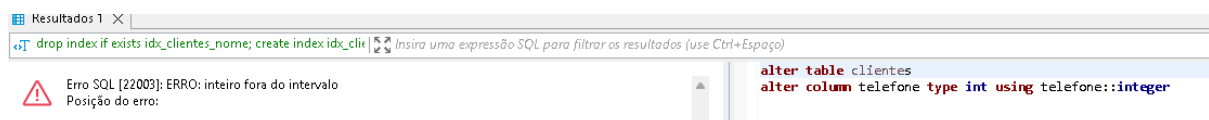
Grade	Plan
1	Seq Scan on clientes (cost=0.00..1.12 rows=1 width=484)
2	Filter: ((nome)::text ~~ 'm%':text)

EXPLAIN depois do índice, mostrando Index Scan.

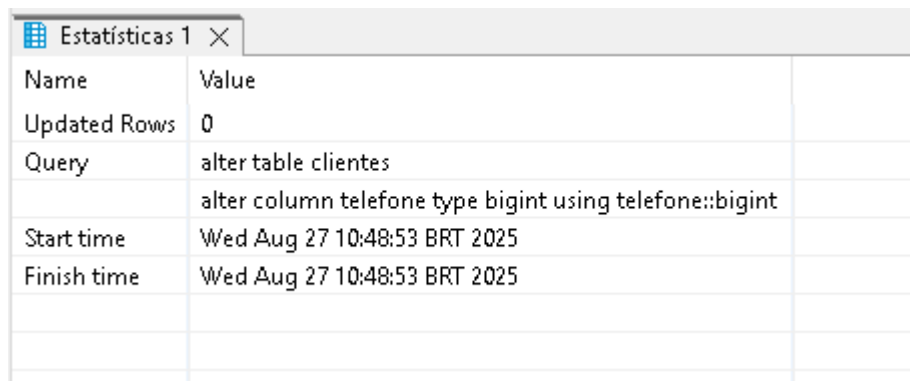
4- Alteração de tipos de dados

Código utilizado:

```
-- Tentativa que deu erro  
alter table clientes alter column telefone type int using telefone::integer;  
  
-- Correção  
alter table clientes alter column telefone type bigint using telefone::bigint;  
alter table clientes alter column cpf type varchar;
```



Erro na alteração de tipo para integer devido ao intervalo dos dados.



The screenshot shows a database statistics window titled 'Estatísticas 1'. The window displays the following statistics:

Name	Value
Updated Rows	0
Query	alter table clientes alter column telefone type bigint using telefone::bigint
Start time	Wed Aug 27 10:48:53 BRT 2025
Finish time	Wed Aug 27 10:48:53 BRT 2025

Alteração de tipo realizada com sucesso para bigint.

Estatísticas 1	
Name	Value
Updated Rows	0
Query	alter table clientes
	alter column cpf type varchar
Start time	Wed Aug 27 10:50:24 BRT 2025
Finish time	Wed Aug 27 10:50:24 BRT 2025

Alteração da coluna CPF para varchar concluída.

5- Criação de usuários e permissões

Código utilizado:

```
do $$
begin
    if not exists (select from pg_catalog.pg_user where username = 'duda') then
        create user duda with password '12345';
    end if;
end
$$;
```


```
do $$
begin
    if not exists (select from pg_catalog.pg_user where username = 'colega') then
        create user colega with password '12345';
    end if;
end
$$;
```

```
grant all privileges on all tables in schema public to duda;
grant select on clientes to colega;
```

Testei os acessos de cada usuário:

clientes 1							
select * from clientes Insira uma expressão SQL para filtrar os resultados (use Ctrl+Espaço)							
	123 id_cliente	A-Z nome	A-Z cpf	A-Z telefone	A-Z email	criado_em	
1	4	carlos pereira	12345678904	11988880004	carlos@email.com	2025-08-25	
2	5	beatriz almeida	12345678905	11988880005	bia@email.com	2025-08-25	
3	6	fernanda costa	12345678906	11988880006	fernanda@email.com	2025-08-25	
4	7	ricardo gomes	12345678907	11988880007	ricardo@email.com	2025-08-25	
5	8	juliana rocha	12345678908	11988880008	juliana@email.com	2025-08-25	
6	9	pedro martins	12345678909	11988880009	pedro@email.com	2025-08-25	
7	10	larissa dias	12345678910	11988880010	larissa@email.com	2025-08-25	
8	2	joão silva	12345678902	[NULL]	joao@email.com	2025-08-25	
9	3	maria santos	12345678903	[NULL]	maria@email.com	2025-08-25	
10	1	teste2	12345678901	[NULL]	ana@email.com	2025-08-25	
11	14	teste	00000000000	[NULL]	[NULL]	2025-08-27	

SELECT funcionando.

Estatísticas 1	
	<p>Erro SQL [23505]: ERRO: duplicar valor da chave viola a restrição de unicidade "clientes_cpf_key"</p> <p>Detalhe: Chave (cpf)=(00000000000) já existe.</p>

```

insert into clientes (nome, cpf) values ('teste', '00000000000');
update clientes set nome = 'erro' where id_cliente = 1;
delete from clientes where id_cliente = 1;

```

INSERT/UPDATE/DELETE dando erro.

clientes 1							
select * from clientes Insira uma expressão SQL para filtrar os resultados (use Ctrl+Espaço)							
	123 id_cliente	A-Z nome	A-Z cpf	A-Z telefone	A-Z email	criado_em	
1	4	carlos pereira	12345678904	11988880004	carlos@email.com	2025-08-25	
2	5	beatriz almeida	12345678905	11988880005	bia@email.com	2025-08-25	
3	6	fernanda costa	12345678906	11988880006	fernanda@email.com	2025-08-25	
4	7	ricardo gomes	12345678907	11988880007	ricardo@email.com	2025-08-25	
5	8	juliana rocha	12345678908	11988880008	juliana@email.com	2025-08-25	
6	9	pedro martins	12345678909	11988880009	pedro@email.com	2025-08-25	
7	10	larissa dias	12345678910	11988880010	larissa@email.com	2025-08-25	
8	2	joão silva	12345678902	[NULL]	joao@email.com	2025-08-25	
9	3	maria santos	12345678903	[NULL]	maria@email.com	2025-08-25	
10	22	teste	00000000011	11999990011	[NULL]	2025-08-27	
11	1	teste2	12345678901	[NULL]	ana@email.com	2025-08-25	

SELECT, INSERT e UPDATE funcionando.

clientes 1 X							
select * from clientes Insira uma expressão SQL para filtrar os resultados (use Ctrl+Espaço)							
	123 id_cliente	A-Z nome	A-Z cpf	A-Z telefone	A-Z email	criado_em	
1	4	carlos pereira	12345678904	11988880004	carlos@email.com	2025-08-25	
2	5	beatriz almeida	12345678905	11988880005	bia@email.com	2025-08-25	
3	6	fernanda costa	12345678906	11988880006	fernanda@email.com	2025-08-25	
4	7	ricardo gomes	12345678907	11988880007	ricardo@email.com	2025-08-25	
5	8	juliana rocha	12345678908	11988880008	juliana@email.com	2025-08-25	
6	9	pedro martins	12345678909	11988880009	pedro@email.com	2025-08-25	
7	10	larissa dias	12345678910	11988880010	larissa@email.com	2025-08-25	
8	2	joão silva	12345678902	[NULL]	joao@email.com	2025-08-25	
9	3	maria santos	12345678903	[NULL]	maria@email.com	2025-08-25	
10	1	teste2	12345678901	[NULL]	ana@email.com	2025-08-25	

DELETE funcionando apenas em registros sem FK.

6- Consultas com JOINS

Código exemplo de JOINS:

-- Inner Join

```
select c.nome, v.id_venda, v.valor_total from clientes c inner join vendas v on c.id_cliente = v.id_cliente;
```

-- Left Join

```
select c.nome, v.id_venda, v.valor_total from clientes c left join vendas v on c.id_cliente = v.id_cliente;
```

-- Right Join

```
select c.nome, v.id_venda, v.valor_total from clientes c right join vendas v on c.id_cliente = v.id_cliente;
```

consultas prints, tirei um print de cada grupo

um com inner join do grupo 1, um com left join do grupo 2, um com right join do grupo 3.

inner:

vendas(+) 1				
select c.nome, v.id_venda, v.valor_total from clientes c innr Insira uma expressão SQL				
	nome	id_venda	valor_total	
1	teste2	1	59,8	
2	joão silva	2	39,9	
3	maria santos	3	89,9	
4	carlos pereira	4	120	
5	beatriz almeida	5	26,9	
6	fernanda costa	6	49,8	
7	ricardo gomes	7	19,9	
8	juliana rocha	8	29,9	
9	pedro martins	9	15,9	
10	larissa dias	10	59,9	

left:

vendas(+) 1				
select v.id_venda, vp.id_produto, vp.quantidade from ven Insira uma expressão SQL				
	id_venda	id_produto	quantidade	
1	1	1	2	
2	2	5	1	
3	3	6	1	
4	4	8	1	
5	5	4	1	
6	6	3	2	
7	7	7	1	
8	8	1	1	
9	9	10	1	
10	10	2	1	

right:

produtos(+) 1 X

select p.nome as produto, c.nome as categoria from prod

	produto	categoria
1	batom vegano	maquiagem
2	base orgânica	maquiagem
3	shampoo sólido	cabelos
4	condicionador natural	cabelos
5	hidratante facial	pele
6	perfume floral	perfumes
7	escova ecológica	acessórios
8	kit skincare	kits
9	esmalte vegano	unhas
10	sabonete infantil	infantil
11	[NULL]	higiene
12	[NULL]	masculino

7- Atualização de valores NULL

Código utilizado:

```
update clientes set telefone = null where id_cliente in (1,2,3);
```

Estatísticas 1 X

Name	Value
Updated Rows	3
Query	update clientes set telefone = null where id_cliente in (1,2,3)
Start time	Wed Aug 27 11:17:29 BRT 2025
Finish time	Wed Aug 27 11:17:29 BRT 2025

Atualização de valores NULL realizada com sucesso.

8- Observações finais

Todas as etapas do exercício foram concluídas com sucesso, incluindo:

- Consultas com LIKE e análise de EXPLAIN
- Criação e uso de índice
- Alterações de tipo de dados com correção de erros
- Criação de usuários e testes de permissões
- Consultas JOIN em 4 grupos
- Atualização de valores NULL