## Banco de dados 2

# Ronaldo Luiz de Araujo Nunes - 123483

## Roteiro 1

## Exercício 1 -

```
1
     CREATE TABLE clientes (
2
     cliente_id SERIAL PRIMARY KEY,
3
     nome TEXT NOT NULL,
     email TEXT NOT NULL
5
     );
6
7
     INSERT INTO clientes (nome, email)
     SELECT
8
9
     'Cliente ' || g AS nome,
      'cliente' || g || '@exemplo.com'
10
     FROM generate_series(1, 1000000) AS g;
11
12
13
```

### Exercício 2 -

```
QUERY PLAN text

Gather (cost=1000.00..15553.43 rows=1 width=43) (actual time=0.878..52.238 rows=1 loops=1)

Workers Planned: 2

Workers Launched: 2

Parallel Seq Scan on clientes (cost=0.00..14553.33 rows=1 width=43) (actual time=7.188..22.889 rows=0 loop...

Filter: (nome = 'Cliente 7000'::text)

Rows Removed by Filter: 333333

Planning Time: 0.068 ms

Execution Time: 52.257 ms
```

### Exercício 3 -

	QUERY PLAN text
1	Index Scan using idx_clientes_nome on clientes (cost=0.428.44 rows=1 width=43) (actual time=0.0320.032 rows=1 loops=
2	Index Cond: (nome = 'Cliente 7000'::text)
3	Planning Time: 1.183 ms
4	Execution Time: 0.047 ms

#### Exercício 4 -

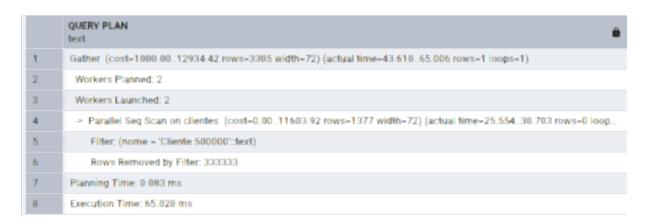
Primeira Consulta (Com Índice): A primeira consulta usou um índice (Index Scan) para encontrar o registro. Isso resultou em um tempo de execução extremamente rápido (0.047 ms) e um custo muito baixo, mostrando a eficiência de usar índices para buscas específicas.

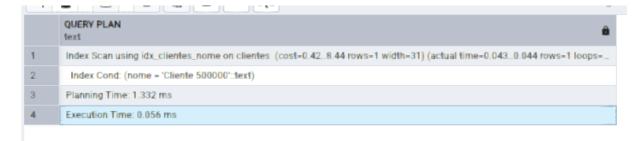
Segunda Consulta (Sem Índice): A segunda consulta realizou uma varredura sequencial completa da tabela (Parallel Seq Scan), lendo todas as 333.333 linhas para encontrar o registro desejado. Isso resultou em um tempo de execução significativamente mais longo (52.257 ms) e um custo muito mais alto, demonstrando a ineficiência de não usar um índice em uma tabela grande.

### Exercício 5 -

Depende da consulta do índice.

#### Exercício 6 -





CREATE INDEX idx\_cidade\_idade ON clientes (cidade, idade);

## C) Análise de Desempenho:

Caso 1: A consulta se encaixa na condição

Consulta: SELECT \* FROM clientes WHERE idade > 60;

Resultado: O otimizador de consultas usará este índice parcial. A busca será muito rápida porque o índice é significativamente menor do que a tabela completa. Ele só contém as chaves para os registros que atendem à condição idade > 60.

Caso 2: A consulta não se encaixa na condição

Consulta: SELECT \* FROM clientes WHERE idade <= 60;

Resultado: O otimizador de consultas não usará este índice. O índice foi criado especificamente para a condição > 60, e não contém informações sobre as linhas com idade <= 60. Para essa consulta, o banco de dados voltará a fazer uma varredura sequencial (Seq Scan) na tabela inteira, o que é muito menos eficiente.