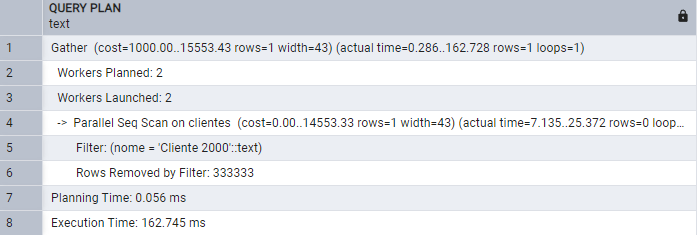
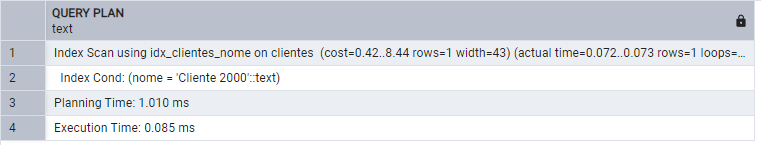
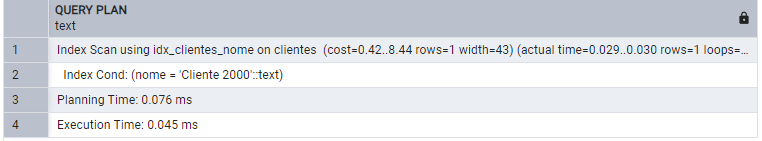
1 - Criar a base e tabela de clientes e popular (Feito)

2 - Realize a consulta sem índice, analisando o plano do Postgres e salvando os dados para comparação futura.



3 - Crie um índice para a tabela clientes e realize novamente a análise do plano de execução.





4 - Faça a comparação entre as duas consultas, descrevendo a diferença entre cada dado.

Sem os index o seu Execution Time foi de 162.745ms já com o index foi de 0,045, isso indica uma melhora de 99,72%

O actual time da sem index foi de 0.072...0.073 já com o index 0.029...0.030 .

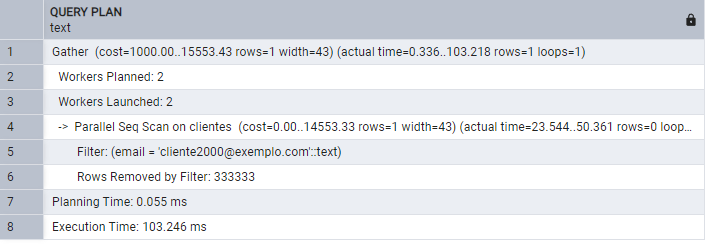
5 - Analise a seguinte afirmação: Para chaves de index diferentes tenho resultados diferentes. Realize os testes com a tabela do exercício. OBS: Para deletar um index já criado utilize o DROP INDEX nome\_do\_index

Email

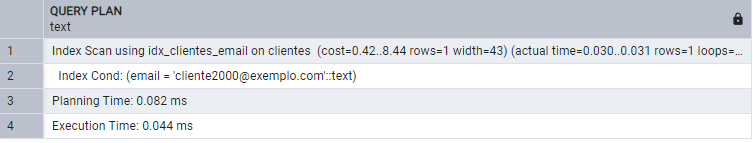
Sem index

EXPLAIN ANALYZE

SELECT \* FROM clientes WHERE email = 'cliente2000@exemplo.com';

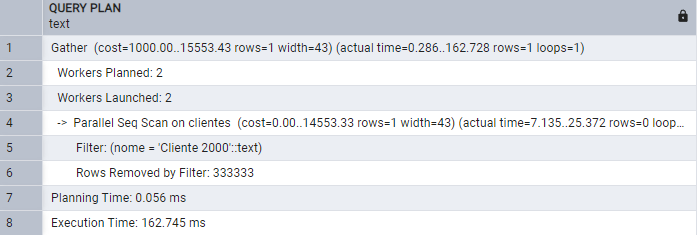


Com index

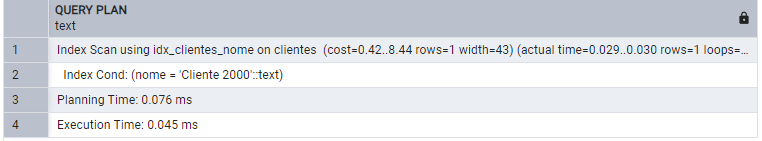
CREATE INDEX idx\_clientes\_email ON clientes(email);  


Nome

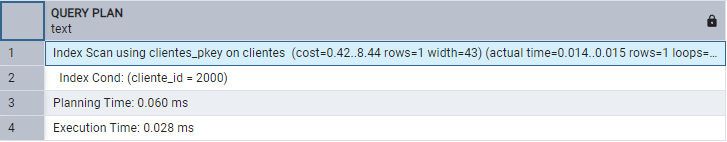
Sem index



Com index



Cliente\_id como e pk já vem com index



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | nome | Nome com index | email | Email com index |
| Execution Time | 162.745 | 0.076 | 103.246 | 0.044 |
| Cost | 1000.00...15553 | 0.42...8.44 | 1000.00...15553 | 0.42.....8.44 |
| Actual Time | 0.286...162.728 | 0.029...0.030 | 0.336...103,218 | 0.030...0.31 |

6 - Crie a seguinte tabela:

Considere o seguinte select:

SELECT \* FROM clientes WHERE cidade = 'Cidade 50' AND idade = 30;

a) Otimize a consulta considerando este caso com mais de uma consição no WHERE.

b) Analise a eficiencia deste índice composto quando ocorre a inversão na ordem das colunas na consulta. c) Crie um índice para apenas uma parte da tabela. Analise o desempenho quando a consulta se encaixa na condição X quando não se encaixa. SELECT \* FROM clientes WHERE idade > 60;