

## Indexação: Planos de Consulta, Seletividade

Nome: Gabriel Stankevix Soares

### Atividade 1

Deve-se criar uma tabela qualquer, e inserir dados aleatórios (acima de 100 mil tuplas). Em seguida:

#### Create Table e função auxiliar

```
CREATE TABLE venda (  
    cod_venda integer primary key,  
    produto varchar,  
    preço float  
);  
  
insert into venda values (1, 'Cerveja', 5.50);  
insert into venda values (2, 'Vinho', 6.50);  
insert into venda values (3, 'Agua', 7.50);  
insert into venda values (4, 'Pinga', 8.50);  
insert into venda values (5, 'Vodka', 9.50);  
  
CREATE OR REPLACE FUNCTION get_random_number(INTEGER, INTEGER) RETURNS  
INTEGER AS $$  
DECLARE  
    start_int ALIAS FOR $1;  
    end_int ALIAS FOR $2;  
BEGIN  
    RETURN trunc(random() * (end_int-start_int) + start_int);  
END;  
$$ LANGUAGE 'plpgsql' STRICT;  
  
do $$  
    begin  
        for i IN 6..300000 LOOP  
            insert into venda values (i,  
                                     (CASE get_random_number(1,5)  
                                      WHEN 1 THEN 'Cerveja'  
                                      WHEN 2 THEN 'Vinho'  
                                      WHEN 3 THEN 'Agua'  
                                      WHEN 4 THEN 'Pinga'  
                                      WHEN 5 THEN 'Vodka' END),  
                                     random() * 10 + 1);  
        end loop;  
    end $$;
```

a) Realizar uma consulta que tenha alta seletividade (calcular a seletividade com a ajuda do select...group by)

#### Consulta base

```
select produto, avg(preço) as media_preço from venda group by produto;
```

explain analyze select produto, avg(preço) as media\_preço from venda group by produto;  
explain select produto, avg(preço) as media\_preço from venda group by produto;

```
Finalize GroupAggregate (cost=5558.15..5558.68 rows=4 width=14) (actual time=155.687..155.692 rows=5 loops=1)
  Group Key: produto
  -> Gather Merge (cost=5558.15..5558.61 rows=4 width=38) (actual time=155.678..162.306 rows=9 loops=1)
    Workers Planned: 1
    Workers Launched: 1
    -> Sort (cost=4558.14..4558.15 rows=4 width=38) (actual time=83.156..83.156 rows=5 loops=2)
      Sort Key: produto
      Sort Method: quicksort Memory: 25kB
      Worker 0: Sort Method: quicksort Memory: 25kB
      -> Partial HashAggregate (cost=4558.06..4558.10 rows=4 width=38) (actual time=83.115..83.116 rows=5 loops=2)
        Group Key: produto
        -> Parallel Seq Scan on venda (cost=0.00..3675.71 rows=176471 width=14) (actual time=0.014..17.341 rows=150000 loops=2)
Planning Time: 0.075 ms
Execution Time: 162.366 ms
```

Seletividade = 1 - (5/300000)

Seletividade = 0.9999833333333333

**b) Realizar uma consulta que tenha baixa seletividade (calcular a seletividade com a ajuda do select...group by)**

select cod\_venda, produto as total from venda where preço > 4.5;  
explain analyze select cod\_venda, produto as total from venda where preço > 4.5;  
explain select cod\_venda, produto as total from venda where preço > 4.5;

```
Seq Scan on venda (cost=0.00..5661.00 rows=195181 width=10) (actual time=0.019..57.180 rows=194856 loops=1)
  Filter: (preço > '4.5'::double precision)
  Rows Removed by Filter: 105144
Planning Time: 0.082 ms
Execution Time: 64.322 ms
```

Seletividade = 1 - (195181/300000)

Seletividade = 0.34939666666666667

**c) Indexar ambas consultas e mostrar os planos de execução com tempos de planejamento e execução das consultas.**

Criar Índice

create index idxproduto on venda (produto);

**Consulta 1**

select produto, avg(preço) as media\_preço from venda group by produto;  
explain analyze select produto, avg(preço) as media\_preço from venda group by produto;  
explain select produto, avg(preço) as media\_preço from venda group by produto;

```

Finalize GroupAggregate (cost=5558.15..5558.68 rows=4 width=14) (actual time=104.566..104.570 rows=5 loops=1)
  Group Key: produto
-> Gather Merge (cost=5558.15..5558.61 rows=4 width=38) (actual time=104.560..111.244 rows=9 loops=1)
  Workers Planned: 1
  Workers Launched: 1
-> Sort (cost=4558.14..4558.15 rows=4 width=38) (actual time=52.041..52.042 rows=5 loops=2)
  Sort Key: produto
  Sort Method: quicksort Memory: 25kB
  Worker 0: Sort Method: quicksort Memory: 25kB
-> Partial HashAggregate (cost=4558.06..4558.10 rows=4 width=38) (actual time=51.995..51.998 rows=5 loops=2)
  Group Key: produto
    -> Parallel Seq Scan on venda (cost=0.00..3675.71 rows=176471 width=14) (actual time=0.015..11.234 rows=150000 loops=2)
Planning Time: 0.077 ms
Execution Time: 111.293 ms

```

## Consulta 2

```

select sum(preço) as total from venda;
explain analyze select sum(preço) as total from venda;
explain select sum(preço) as total from venda;

```

```

Seq Scan on venda (cost=0.00..5661.00 rows=195181 width=10) (actual time=0.017..35.985 rows=194856 loops=1)
  Filter: (preço > '4.5'::double precision)
  Rows Removed by Filter: 105144
Planning Time: 0.083 ms
Execution Time: 40.643 ms

```