

Nome: Gabriel Stankevix Soares

Tarefa Semanal

1. Criar uma tabela contendo, dentre seus atributos, um atributo que possa ser particionado/categorizado, por exemplo, na aula vimos cidades e faixas de endereços IP.

Create Table e função auxiliar

```
CREATE TABLE venda (  
    cod_venda integer primary key,  
    produto varchar,  
    preço float  
);
```

2. Popular a tabela com dados aleatórios.

```
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 $$  
begin  
    for i IN 6..300000 LOOP  
        insert into venda values (i,  
            (CASE get_random_number(1,5)  
                WHEN 1 THEN 'Cerveja'  
                WHEN 2 THEN 'Vinho'  
                WHEN 3 THEN 'Agua'  
                WHEN 4 THEN 'Pinga'  
                WHEN 5 THEN 'Vodka' END),  
            random() * 10 + 1);  
    end loop;  
end $$;
```

3. Indexar o predicado do atributo particionado e mostrar uma consulta sobre ele usando o índice particionado. O importante é exercitar o uso da cláusula "where" no create index.

```
explain select * from venda where produto = 'Pinga';
```

QUERY PLAN		
text		
1	Seq Scan on venda (cost=0.00..5661.00 rows=75140 width=18)	
2	Filter: ((produto)::text = 'Pinga'::text)	

Criando o índice

```
CREATE INDEX idxcod_venda ON venda(cod_venda) WHERE produto='Pinga';
```

```
explain select * from venda where produto = 'Pinga';
```

[Data Output](#) [Explain](#) [Query History](#) [Notifications](#) [Messages](#)

QUERY PLAN		
text		
1	Index Scan using idxcod_venda on venda (cost=0.29..2441.10 rows=75140 width=18)	