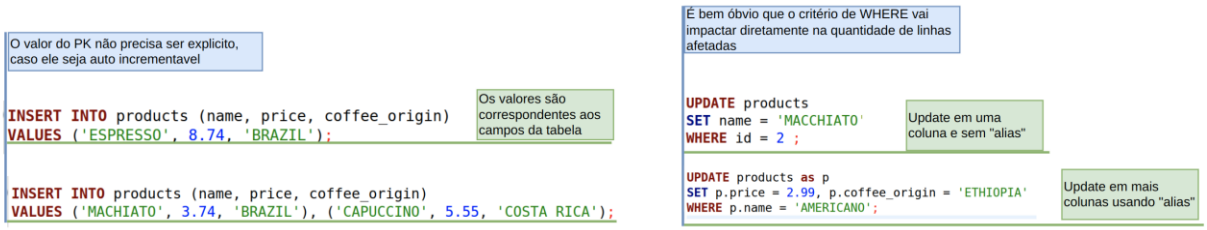
DML – Inserir, Atualizar e Deletar

DDL é como você se refere a **comandos SQL que tem como função manipular dados de tabelas**. Por exemplo: *inserir, buscar, filtrar, remover*:




```
INSERT INTO products (name, price, coffee_origin)
VALUES ('ESPRESSO', 8.74, 'BRAZIL');

INSERT INTO products (name, price, coffee_origin)
VALUES ('MACHIATO', 3.74, 'BRAZIL'), ('CAPUCCINO', 5.55, 'COSTA RICA');
```

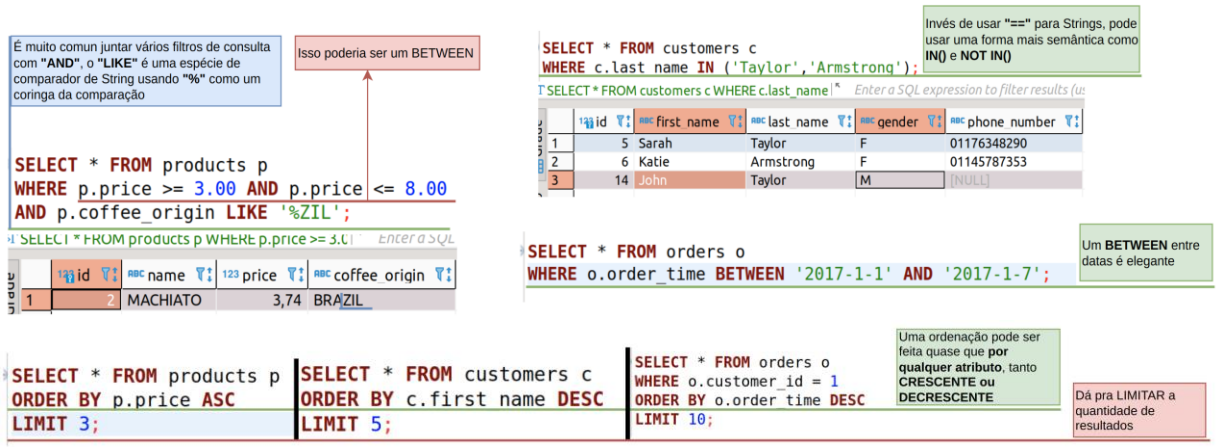
```
UPDATE products
SET name = 'MACCHIATO'
WHERE id = 2 ;

UPDATE products as p
SET p.price = 2.99, p.coffee_origin = 'ETHIOPIA'
WHERE p.name = 'AMERICANO';
```

A sintaxe de deleção das linhas é bem simples **"DELETE FROM <> WHERE <>",** essas operações também podem ter mais critérios encadeados usando **AND, LIKE, etc...**

DML – Consultas Básicas

Consultas tem o objetivo de retornar dados específicos, os **critérios são os mais variados possíveis.** Da pra ordenar, selecionar o maior valor, somar, usar operadores lógicos e etc...:




```
SELECT * FROM products p
WHERE p.price >= 3.00 AND p.price <= 8.00
AND p.coffee_origin LIKE '%ZIL';
```

	id	name	price	coffee_origin
1	2	MACHIATO	3,74	BRAZIL

```
SELECT * FROM customers c
WHERE c.last_name IN ('Taylor','Armstrong');
```

	id	first_name	last_name	gender	phone_number
1	5	Sarah	Taylor	F	01176348290
2	6	Katie	Armstrong	F	01145787353
3	14	John	Taylor	M	[NULL]

```
SELECT * FROM orders o
WHERE o.order_time BETWEEN '2017-1-1' AND '2017-1-7';
```

```
SELECT * FROM products p
ORDER BY p.price ASC
LIMIT 3;
```

```
SELECT * FROM customers c
ORDER BY c.first_name DESC
LIMIT 5;
```

```
SELECT * FROM orders o
WHERE o.customer_id = 1
ORDER BY o.order_time DESC
LIMIT 10;
```

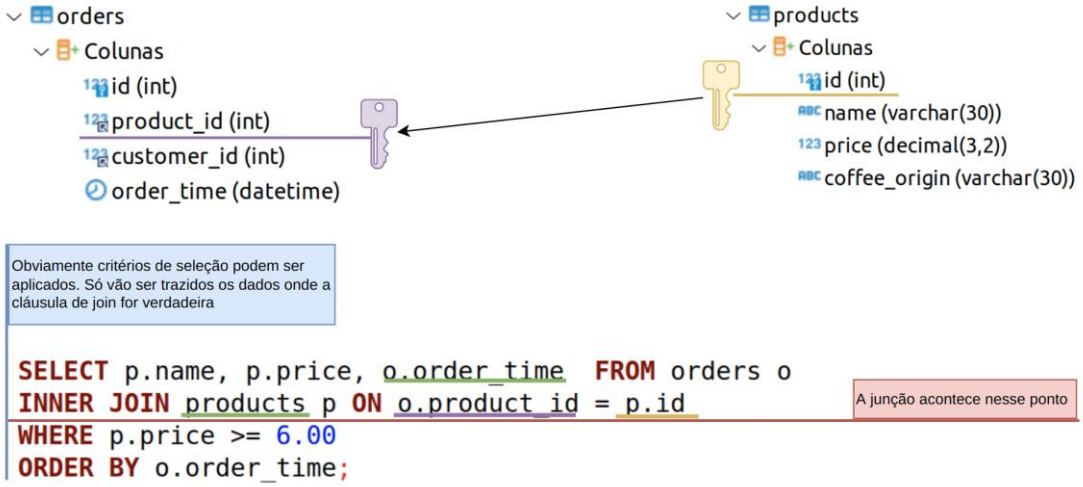
O **distinct** não entrou em exemplo, mas o uso dele é mto fácil também. É só aplicar o DISTINCT em uma coluna específica e você vai ter todos os valores distintos da coluna, também dá pra usar como filtro em where e etc...

DML – Consultas em múltiplas Tabelas

A sintaxe básica que te permite juntar dados de múltiplas tabelas é o "JOIN", obviamente **só dá pra juntar tabelas que possuem uma relação através das chaves.**

INNER JOIN: Retorna os dados que dão "match" nas duas (pode ser mais, é só ir dando JOIN) tabelas, esse match é relacionado a tabela 2 possuir valor atrelado a um campo chave na tabela 1 e

vice versa:



LEFT JOIN: Traz **TODOS OS DADOS da tabela na esquerda (pode entender como a primeira tabela referenciada)** e os **dados que combinam da tabela direita**. É tipo tu pegar todos os dados da tabela 1 e os que dão o match na tabela 2.

RIGHT JOIN: É igual o left join, a diferença é que a tabela que vai retornar todos os dados é a 2(da direita).