Nesse vídeo, vamos nos aprofundar no LEFT JOIN, que é o comando de junção de tabelas que mostrará **todos** os elementos da tabela **à esquerda** do JOIN(ou seja, a que vem antes do JOIN) e somente os correspondentes da tabela à direita.

Vamos abrir o MySQL Workbench, criar um novo script e começar com uma seleção que trará a contagem de clientes cadastrados:



Lembrete: quando omitimos os campos selecionados e somente usamos uma fórmula, não há necessidade de usar o GROUP BY. É o que acabamos de fazer com a fórmula COUNT.

O retorno revela que existem 15 clientes cadastrados. Em seguida, vamos consultar para quantas pessoas foram emitidas notas fiscais, verificando a quantidade de CPFs diferentes que aparecem na tabela de notas fiscais:

Texto

O conteúdo gerado por IA pode estar incorreto.

Ao contar os registros, saberemos que apenas 14 clientes receberam nota fiscal. Ou seja, dos 15 cadastrados, 1 deles nunca comprou suco de frutas na nossa empresa.

Para descobrir quem é esse cliente, primeiro vamos rodar uma consulta com INNER JOIN para descobrir quais registros apresentam correspondências no campo "CPF":

Texto

O conteúdo gerado por IA pode estar incorreto.

Ou seja, o retorno mostrará os CPFs e os nomes dos clientes para os quais foram emitidas notas - são as 14 pessoas que vimos anteriormente. No entanto, substituindo o INNER JOIN pelo LEFT JOIN, o resultado será diferente:

Interface gráfica do usuário, Texto

O conteúdo gerado por IA pode estar incorreto.

Agora, veremos **todos** os elementos da tabela de clientes e apenas os correspondentes da tabela de notas fiscais. Analisando o resultado, encontraremos o cliente Fábio Carvalho que tem um campo nulo, ou seja, que nunca comprou na empresa, então nunca recebeu nota fiscal e, consequentemente, seu CPF não tem consta na tabela de notas fiscais.

Desse modo, se o gerente da empresa de sucos nos pedisse para investigar quais clientes cadastrados nunca realizaram uma compra, uma maneira de encontrar a resposta é filtrar os registros que apresentam o campo B.CPF nulo (no caso, apenas o Fábio):

Texto

O conteúdo gerado por IA pode estar incorreto.

IS, em inglês, significa "é/está". Logo, no comando WHERE B.CPF IS NULL, buscamos campos que têm valor *null*.

Lembrando que sempre temos a opção de incrementar nossos filtros. Um exemplo seria refinar a busca pelo ano da data da venda, que é uma informação que encontramos na tabela de notas fiscais:

Texto

O conteúdo gerado por IA pode estar incorreto.

Nesse caso, utilizamos o operador AND, então nosso retorno será vazio, porque não há registros que atendam às duas condições **ao mesmo tempo**.

Quanto ao RIGHT JOIN, sabemos que ele tem quase a mesma mecânica do LEFT JOIN, exceto que trará todos os elementos da tabela **à direita** do comando JOIN e somente os correspondentes da outra tabela. Para fazer uma demonstração, podemos realizar a mesma consulta que fizemos anteriormente, invertendo os nomes das tabelas:

Interface gráfica do usuário, Texto

O conteúdo gerado por IA pode estar incorreto.

Apesar da consulta ligeiramente diferente, como invertemos as tabelas, o resultado é o mesmo que obtivemos com o LEFT JOIN.