Falaremos, a seguir, de uma estrutura importante no MySQL chamada **visão**, que é uma tabela lógica, resultado de uma consulta. Ao realizar uma *query* (seja simples ou complexa) podemos salvá-la com um nome (por exemplo, "V") e mais tarde referenciar essa visão como se fosse uma tabela (SELECT \* FROM V).

As visões (ou *views*) são bastante úteis quando precisamos disponibilizar parte do nosso banco de dados para uma pessoa externa, pois criamos uma tabela lógica com limites que garantam a segurança das informações e concedemos acesso somente a essa visão.

Como a *view* é uma consulta, ela tem um custo. Toda vez que referenciamos uma visão, pelos bastidores do banco de dados estamos executando uma seleção. Assim sendo, se a visão for um SELECT muito complexo, ao acessá-la o desempenho pode ser prejudicado.

Como exemplo, usaremos uma tabela que tem o campo "X" com letras e o campo "Y" com números:

Vamos supor que fizemos a seguinte seleção, que agupa os valores iguais de "X" e traz a soma dos respectivos "Y":



E obtemos um retorno que chamaremos de tabela 2:

Daremos um nome a esse SELECT que executamos, criando uma visão chamada "VW\_VIEW". Então, se rodarmos SELECT \* FROM VW\_VIEW, teremos como retorno o resultado de SELECT X, SUM(Y) as NEW\_Y FROM tab1 GROUP BY X (a tabela 2). Ou seja, é como se essa *view* fosse uma subconsulta.

Com essa visão podemos, por exemplo, fazer um JOIN com outra tabela ou até com outras visões. Testaremos esse conceito usando a seguinte tabela (que chamaremos de tabela 3):

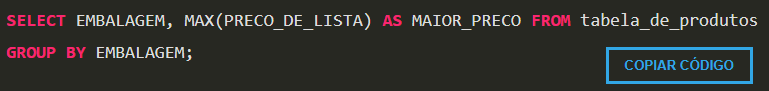
Vamos fazer a seguinte consulta com INNER JOIN:

Texto

O conteúdo gerado por IA pode estar incorreto.

E o retorno será o seguinte:

Vamos ao MySQL Workbench para praticar no nosso banco de dados. Criaremos um novo script e repetiremos a consulta que fizemos no vídeo passado, selecionando os maiores preços de acordo com as embalagens:



E, novamente, usaremos esse SELECT como uma subconsulta para ver apenas os preços máximos que são maiores ou iguais a 10:

Texto

O conteúdo gerado por IA pode estar incorreto.

Até este ponto é o que fizemos no vídeo anterior. Agora, criaremos uma *view* com o conteúdo dessa subconsulta. No painel à esquerda do programa, dentro da base "sucos\_vendas" que está em negrito, vamos clicar em "*Views*", com o botão direito do mouse. Em seguida, selecionaremos "*Create View...*" e uma nova aba será aberta. Nela, vamos digitar o seguinte código:

Texto

O conteúdo gerado por IA pode estar incorreto.

Dessa forma, estamos criando uma visão com nome "VW\_MAIORES\_EMBALAGENS".

É padrão usar as letras "VW" no início dos nomes das *views*, essa prática facilita a identificação das visões na hora de programar.

A seguir, clicaremos em "*Apply*", depois em "OK". Uma caixa de diálogos será aberta, mostrando o script que o programa vai executar. Note que o código está ligeiramente diferente, em vez do CREATE VIEW que digitamos, temos CREATE OR REPLACE VIEW. Essa mudança significa que o MySQL vai criar a visão **ou** subtituí-la, caso já exista. Vamos pressionar "*Apply*" mais uma vez e, então, "*Finish*".

Agora, no painel à esquerda, é possível expandir o subgrupo *Views* e, dentro dele, temos a visão que acabamos de criar. Analisando o conteúdo de "vw\_maiores\_embalagens", veremos as colunas "EMBALAGEM" e "MAIOR\_PRECO". Ou seja, a visão tem um comportamento igual ao de uma tabela.

Voltando ao nosso script, agora que criamos a visão podemos refazer a última seleção de um jeito alternativo, substituindo a subconsulta pela nova *view*:

Texto

O conteúdo gerado por IA pode estar incorreto.

Também temos a opção de fazer, por exemplo, um JOIN com informações da tabela de produtos e dados da visão:

Texto

O conteúdo gerado por IA pode estar incorreto.

Trata-se de um INNER JOIN entre uma tabela e a *view* que criamos que, nos bastidores, está executando o agrupamento dos maiores preços de cada tipo de embalagem.

Com essa seleção, verificamos no primeiro registro, por exemplo, que o suco "Sabor da Montanha - 700 ml - Uva" em garrafa custa R$6,30 e o produto mais caro vendido em garrafa é R$13,31. Para descobrir qual é o item em garrafa mais caro, basta encontrar o registro em que a embalagem é garrafa e o preço de lista também é R$13,31 - é o "Festival de Sabores - 2 Litros - Açaí".

Podemos, inclusive, criar um indicador de porcentagem para saber o quão mais barato está cada produto, comparado ao valor da coluna "MAIOR\_PRECO":

Texto

O conteúdo gerado por IA pode estar incorreto.

Dessa forma, concluímos que "Sabor da Montanha - 700 ml - Uva" vendido em garrafa é 52% mais barato que a garrafa mais cara. Já "Festival de Sabores - 2 Litros - Açaí" é 0% mais barato, pois é justamente a garrafa mais cara.

Conhecendo visões, portanto, já começamos a fazer alguns relatórios, até mesmo calculando indicadores como percentual de preços. Juntando vários conceitos de SQL, estamos adquirindo noções de como construir relatórios para os clientes a partir da análise de informações dos bancos de dados.