

5.3 - Conhecendo SQL (parte 1)

00;00;00;07 - 00;00;30;04

Olá pessoal! Em umas aulas anteriores nós falamos que geralmente a linguagem utilizada para consultar e manipular os dados é o SQL. Agora chegou a hora de falarmos mais sobre essa linguagem. Para iniciar a conversa, o SQL vem de Structured Query Language e é uma linguagem de programação projetada para gerenciar e manipular bancos de dados relacionais. Ele fornece uma maneira padronizada de executar operações como consultas, inserções, atualizações e exclusões de dados em banco de dados.

00;00;30;14 - 00;00;55;07

SQL é amplamente utilizado em banco de dados relacionais para realizar uma variedade de tarefas e ela tem uma parte legal que divide os comandinhos da linguagem de acordo com a finalidade dela. Assim, a gente consegue aprender todas, mas depois focar no que a gente for utilizar mais em nosso dia a dia de trabalho. E as divisões são assim: a gente começa pelo DDL, que é "Data Definition Language", que basicamente são comandos para incluir, criar e modificar uma tabela, o create table, alter table e drop table.

00;00;55;18 - 00;01;21;09

Depois a gente tem DML, de "Data Manipulation Language", que são comandos de atualização, seleção, consulta e remoção dos dados de uma tabela, como o select, o update e o delete. E depois a gente tem o DCL, Data Control Language, comandos mais avançados, principalmente para permissões de usuários. Óbvio que a gente tem mais aí, mas esses são os três principais.

00;01;21;23 - 00;01;46;18

Os comandos dentro do DML, principalmente quando se fala da parte de select, são os que mais utilizaremos no nosso dia a dia como analista de dados. Afinal, precisamos consultar os dados que queremos para poder analisá-los. Mas vamos dar uma olhadinha nos comandos de DDL, porque aqui vamos precisar criar a nossa tabela de municípios e vai ser importante no nosso dia a dia para entender como algumas tabelas foram criadas e relacionadas quando não houver diagrama ou documentação sobre os dados.

00;01;46;18 - 00;02;23;00

Uma informação importante, uma construção de comando em SQL é chamada de Query e, em termos simples, uma Query é uma pergunta que você faz ao banco de dados para obter informações específicas ou realizar operações sobre os dados armazenados nele. Voltando lá no DBEaver, da tabela de álbum que exploramos na última aula, vamos ver qual foi a Query utilizada para criar a tabela e para isso a gente vai lá na tabela de álbum, que está aqui já aberta, aqui em cima na aba, e aí a gente clica aqui em DDL.

00;02;23;23 - 00;02;54;09

Aqui... Aparece pra nós o script em SQL para criação dessa tabela. Eu acho isso muito legal, gente. Bom, podemos ver que o primeiro comando é o create table e aí o nome do álbum, ou seja, em livre tradução aqui: crie a tabela álbum. Fácil, né? É quase como dar uma ordem ao banco. E aí dentro dos parênteses aqui temos as colunas do álbum que são: algum id, título, artes id e o column 1. E seus tipos: inteiro, varchar, inteiro e o blob.

00;02;54;20 - 00;03;19;11

O varchar 160 aqui vamos resumir que é para um texto que tem no máximo 160 caracteres; o inteiro, que é para números inteiros. e o blob, que é um objeto, podendo ser uma foto, uma lista ou qualquer tipo não simples de variável. Também estamos falando que todas as colunas precisam ser obrigatoriamente preenchidas com esse "not null".

00;03;20;00 - 00;03;48;03

Apenas a coluna Column 1 que tem null, ou seja, não está o "not null": significa que ela pode ter nulos. E por fim, definimos duas restrições: os constraint. Uma é a nossa chave primária, o primary key, que, como já vimos, define qual coluna vai ser o identificador único de cada registro, nesse caso, o Album Id; e a chave estrangeira, que é o Foreign Key, que define o relacionamento dessa tabela com outra quando existe.

00;03;48;13 - 00;04;09;08

Nesse caso, temos a Artist Id, que vem da tabela Artist e se liga com a coluna também chamada Artist Id de lá, ok? Artist, Artist Id de lá. Pronto, conseguimos ler o nosso DDL. Depois de traduzir algumas palavras, vemos que é quase como dar uma instrução escrita mesmo, não é? Vamos agora construir as nossas tabelas de município?

00;04;09;17 - 00;04;44;06

Vejam, eu já me adiantei e trouxe um diagrama aqui de como vai ficar depois que a gente criava as nossas tabelas. Eu peço para vocês ignorarem os nomes aqui de colunas, porque a gente vai modificar, ok? As tabelas de gerência, região e municípios brasileiros vão se relacionar pela coluna de região, ok? Tem região aqui e região aqui. E as tabelas de municípios brasileiros com município status vão se relacionar pela coluna de município id. Município id aqui e município id aqui. Sempre que for preciso criar tabelas que se relacionem, eu aconselho a fazer um diagrama.

00;04;44;13 - 00;05;12;25

Pode ser um papel mesmo escrito com um lápis só pra ter uma melhor visualização desses relacionamentos e quais colunas que vão ser chaves. Ok, então vamos pra o Dbeaver criar nossa conexão. Vamos clicar aqui em Arquivo e clicar em Novo. Nessa nova aba aqui que vai se abrir, a gente vai clicar aqui em Dbeaver, abrir as opções, clicar em Conexão com o banco de dados e depois em Avançar.

00;05;12;25 - 00;05;37;27

E na próxima aba que abrir vamos selecionar qual o banco de dados que vamos usar. Para esse curso vamos selecionar o SQLite, que é um sistema de gerenciamento de banco de dados relacional que implementa uma versão mínima do SQL. Beleza, vamos clicar aqui no SQLite e depois clicar em Avançar. Na próxima aba que aparecer, nós vamos clicar em Criar aqui para criar um novo arquivo de banco de dados.

00;05;38;18 - 00;06;00;21

A gente vai selecionar aqui onde no nosso computador que vai ficar esse arquivo e eu gostaria que vocês colocassem em lugar que vocês não vão esquecer, porque depois a gente vai usar esse arquivo, beleza? Eu vou colocar aqui em Documentos que eu sei que é fácil de encontrar. Está em Documentos e vamos chamar esse arquivo de Status Brasil, beleza?

00;06;00;29 - 00;06;33;24

E aí a gente clica em salvar. Ok. E vamos clicar em Concluir. E aí a gente pode ver que já foi criado aqui o nosso Banco de Dados Status Brasil. Porém, se a gente clicar aqui pra poder expandir e clicar em tabelas, não vai ter nenhuma, pois ainda não criamos nada. Então vamos lá criar nossas tabelas. Vou fechar aqui esse... fechar não, vou encolher aqui esse "sample" só pra gente poder conseguir visualizar melhor o nosso Status Brasil.

00;06;34;10 - 00;07;03;04

Vamos clicar aqui em Editor SQL, na barra de ferramentas e vamos clicar em Novo Script SQL. E aí vai abrir aqui uma nova aba onde a gente pode escrever o nosso código em SQL. Como vimos, começamos nosso código com o comando básico de SQL, que é o Create Table. Então vamos digitar: create table. A próxima coisa é dar um nome para essa tabela.

00;07;03;18 - 00;07;34;03

Vamos começar com a tabela de municípios brasileiros, então vamos: create table municípios brasileiros. Ok? Legal, agora a gente precisa criar as colunas dessa tabela. Então, continuando nosso código, vamos criar as colunas assim: vamos colocar aqui os parênteses, porque a gente está criando a tabela e agora a gente vai criar as colunas dessa tabela. Então a gente precisa colocar os parênteses para falar que as colunas estão dentro dessa tabela, ok?

00;07;34;20 - 00;08;13;23

E para essa tabela a gente tem quatro colunas: cidade, estado, região e município id. Ok? E aí a gente tem aqui os nomes das colunas e gente precisa informar qual o tipo de cada coluna. Por exemplo, cidade vai ser do tipo texto: nvarchar. O estado também é nvarchar e região também. Vou copiar e colar que vai ficar mais fácil, né gente?

00;08;14;24 - 00;08;45;01

E o município id é inteiro: interger. Bom, aí a gente já colocou o nome de cada coluna e o tipo. Eu até escrevi "interger" com um R a mais. Pronto. beleza. E aí a gente precisa colocar vírgula depois de cada uma: cidade nvarchar vírgula vírgula vírgula. Pronto! A última não precisa ter vírgula. Vejam bem, nós abrimos os parênteses para dizer que tudo que está dentro dos parênteses é relacionado à criação dessa tabela.

00;08;45;23 - 00;09;02;12

Então nós colocamos ali cidade seguida do nvarchar. Isto é, a cidade é o nome da nossa coluna e o nvarchar o tipo de dado dessa coluna, que a coluna aceita. Ok? Além de dizer que essas colunas são do tipo de texto, nós podemos ser mais restritivos e dizer qual o máximo de caracteres que essa coluna aceita por cada entrada.

00;09;02;29 - 00;09;31;19

Então, por exemplo, aqui em cidade eu posso colocar, depois do nvarchar, o 50, que é pra dizer que cada entrada nessa coluna vai aceitar no máximo 50 caracteres. O estado, por exemplo, eu quero que as entradas sejam do tipo em sigla (SP, RJ), então vou limitar aqui a entrada por 2 caracteres. Na região a gente pode colocar, limitar por 10 caracteres, por exemplo.

00;09;33;01 - 00;09;55;19

Então tá, aqui a gente já colocou as colunas, colocamos os tipos e limitamos ali para as colunas, que são do tipo de texto. Outro acréscimo que podemos fazer é colocar que os itens precisam ser obrigatoriamente preenchidos. Para isso, nós usamos o comando Note Null, que é basicamente não nulo, ou seja, não aceita nulos nessa coluna.

00;09;56;04 - 00;10;37;15

Então o nosso código pode ficar assim... Aqui em cidade a gente coloca Note Null e em id também, né? E, gente, a produção me informou que Centro-Oeste tem mais de dez caracteres e a gente tinha limitado aqui a região por 10. Então vamos mudar, vamos colocar 20 para poder acolher todo mundo? Agora nós precisamos definir nossa chave primária, que basicamente é a coluna que vamos usar para conectar com outra tabela. Nós temos o município id, que é uma coluna com números únicos.

00;10;37;15 - 00;11;06;11

Então vai ser essa coluna que vamos definir como chave primária. Aí aqui a gente tem município id, a gente já definiu algumas coisas e agora a gente coloca o comando Primary Key. E isso já diz que essa vai ser a nossa coluna de chave primária, beleza? Para finalizar, nós podemos adicionar na linha em que criamos essa coluna, o comando dizendo que essa coluna, além de chave primária, ela é do tipo auto incremental e o comando é auto increment.

00;11;06;11 - 00;11;30;11

Que basicamente gera um novo número para cada novo registro que adicionarmos nessa tabela. Assim, garantimos que os valores do id sejam únicos e que a gente nem precisa preencher essa coluna. Então a gente coloca aqui auto increment. Então vamos lá, gente...

00;11;30;11 - 00;12;00;08

Nessa coluna em específico, município id, a gente já falou que ela é do tipo inteiro, ela vai ser com números, ela não pode ficar vazia, então ela é note null, ela é a chave primária (primary key) e ela é do tipo auto increment, auto incremental. Precisa ser nessa ordem, viu, gente? Se vocês colocarem auto increment antes da chave primária, vai dar erro quando a gente executar. Bom, e pra criar a tabela de fato, nós clicamos aqui no símbolo de executar pra executar a instrução SQL. Aqui... Aqui embaixo já aparece, não deu erro, tudo certo.

00;12;00;26 - 00;12;25;12

Mas a gente precisa desconectar e conectar de novo o banco de dados pra poder atualizar e tabela aparecer aqui no cantinho esquerdo pra gente. Então a gente vai aqui em Status Brasil, clica com o botão direito, vai em Desconectar. A gente salva aqui as alterações do nosso script, ok? Bom, aí a gente vai aqui expandir e aqui está a nossa tabela criada: municípios brasileiros.

00;12;25;12 - 00;12;47;02

Se a gente clicar aqui nela, a gente consegue ver as colunas aqui que gente criou, os tipos, a gente consegue ver que o município aí é do tipo auto increment. Se a gente for aqui em chave única, a gente consegue ver também que já está aqui a chave primária. Ok? Bom, agora a gente vai criar a nossa tabela de município status. Ok? Pausa o vídeo aí.

00;12;47;16 - 00;13;09;14

Uma dica, né? Pausa o vídeo aí é tenta fazer o código da criação dessa tabela. Lembrando que as colunas são colunas de status id, população residente, IDHM Ranking, educação, renda e município. A gente pode até ir lá no diagrama para vocês verem. A gente criou essa tabela aqui, municípios brasileiros e agora a gente vai criar essa município status.

00;13;09;14 - 00;13;22;10

Tentem fazer aí, criar essa tabela. Essa tabela tem essas colunas que a gente já falou e depois volta pra gente poder comparar se está tudo certo e tudo mais. Ok?