

# BANCO DE DADOS

## CONSULTAS ESTRUTURADAS

Quando falamos em programação nos bancos de dados, estamos tratando respectivamente da programação, que consiste em um conjunto de técnicas para criar procedimentos estruturados que permitem aos computadores executarem tarefas desejadas, e dos esquemas de bancos de dados relacionais. Nesses esquemas, temos dados armazenados em tabelas, que podem ser detectados de forma única e onde alguns campos podem ser nulos. Consultas são resultados de produtos cartesianos das tabelas selecionadas no comando FROM, realizadas pelas conjunturas do comando WHERE, mostrando os resultados nos campos do comando SELECT.

## ETAPAS PARA O DESENVOLVIMENTO DE UM BANCO DE DADOS

Quando vamos desenvolver um banco de dados, temos que, antes de tudo, pensar na forma de sua estrutura, de suas tabelas e seus relacionamentos. Para que isso possa acontecer, devemos compreender o problema que o banco de dados resolverá e quais são seus entraves, assim ele pode fornecer todos os dados necessários para atender suas metas. Depois, devemos apontar os elementos fundamentais do banco de dados, que nesse caso consistem nas suas tabelas. Em seguida, presumindo que todas as tabelas estejam devidamente determinadas, modelaremos a sua estrutura. Para encerrar, escolhemos os tipos de dados e estabelecemos relacionamentos.

## CONJUNTOS DE CARACTERES

Quando estamos desenvolvendo um banco de dados, sempre devemos prestar atenção na sua abrangência de uso, pois os dados que nele serão armazenados deverão respeitar as regras de gramática ou representação de cada país. Os sistemas gerenciadores de bancos de dados têm comandos especiais para atender estas questões. Esses comandos consistem no CHARSET, que denomina um conjunto de caracteres e determina a forma como os mesmos são representados no banco de dados, e no COLLATION que aponta as regras para comparação de caracteres.

## TRABALHANDO COM DADOS E TABELAS

Todo projeto precisará ser ajustado conforme o tempo. Quando ele está começando, isso não é um problema, pois queremos colocar à prova sua estrutura. Porém, os projetos nem sempre estão nessa fase e em muitos desses casos, os bancos de dados podem ser alterados. Quando vamos trabalhar com os dados existentes em uma tabela, podemos usar alguns comandos. Um INSERT acrescenta novos registros, um UPDATE muda registros e um DELETE remove registros.

## ALTERANDO COLUNAS

Dentro do comando ALTER TABLE, podemos usar outros comandos que nos permitem renomear ou reordenar colunas e mudar os aspectos de uma tabela. O comando CHANGE pode renomear uma coluna e também mudar seus aspectos. Quando usado em conjunto com o comando FIRST ou com o comando AFTER, podemos reordenar as colunas da tabela. O comando MODIFY é mais conveniente para mudar os aspectos sem mudar o nome de uma coluna. O comando RENAME COLUMN é mais conveniente para mudar o nome de uma coluna.

## REMOVENDO TABELAS

Quando um banco de dados tem sua estrutura determinada, com todos os seus campos, tipos de dados e chaves, ele está apto a ter dados adicionados, alterados ou removidos. Porém, antes devemos fazer CONSTRAINTS, garantindo relacionamentos íntegros entre as tabelas, mantendo a integridade dos dados e evitando entradas errôneas. Para remover uma ou mais tabelas ou seus conteúdos, os comandos DROP e TRUNCATE podem ser usados, respectivamente.

## CONSULTAS AVANÇADAS

Estudamos formas de consultas em tabelas únicas, mas essas formas não nos permitem fazer o mesmo em diversas tabelas. Para que possamos fazer isso, precisamos inserir alguns registros em ambas as tabelas e usar o comando **SELECT**, de forma conjunta com o comando **JOIN**. Dessa forma, podemos juntar duas ou mais tabelas, apontando os seus campos correspondentes. Nós podemos também usar os comandos **LEFT JOIN** e **RIGHT JOIN**.

## ANINHAMENTO DE CONSULTAS

Anteriormente, entendemos como ocorrem as consultas em duas ou mais tabelas, mas também podemos fazer consultas aninhadas, compostas por comandos **SELECT**, **FROM** e **WHERE**, usados dentro de outro **SELECT**. Assim, nós podemos comparar conjuntos de dados, usando conectivos como **IN**, que faz testes num determinado conjunto de dados, e **NOT IN**, que busca determinada ausência num conjunto de valores. Essas consultas, não precisam necessariamente, ser feitas em duas tabelas relacionadas, sendo possível usar esse recurso em apenas uma tabela.

## RECURSOS AVANÇADOS

Assim como nos navegadores, os bancos de dados também possuem recursos que promovem maior velocidade às buscas de informações. Quando é executada uma seleção, ocorre uma checagem de correspondência em um ou mais campos, nesse processo em tabelas com um grande número de registros, o tempo de espera tende a ser elevado. Para sanar esse problema e conter a taxa de processamento, podemos usar os comandos **INDEX** ou **VIEW**, que quando executados dentro de um comando **SELECT**, retornam as tabelas de forma mais veloz.

## CONTROLE TRANSACIONAL

Os sistemas gerenciadores de bancos de dados devem ter mecanismos que possam garantir o controle de todos os dados arquivados no mesmo. As transações são consideradas conjuntos de operações, em que se pode acessar mudar ou remover dados em uma ou mais tabelas.

## COMMITTS

Quando uma transação se completa, um novo estado é automaticamente criado. **COMMITTS** vem geralmente em modo automático, sendo conhecidos como **AUTOCOMMITTS**. Nesse caso, todas as mudanças são armazenadas, sem nenhum comando dos usuários. Quando colocamos seu valor em zero, estamos passando os **COMMITTS** para o modo manual.

## ROLLBACKS E SAVEPOINTS

Quando queremos reverter às transações, podemos usar o comando **ROLLBACK**, retornando aos estados anteriores. Mas para usarmos esse comando, devemos também, fazer um **SAVEPOINT**.

## PROCESSOS AUTOMATIZADOS

Temos recursos que servem para automatizar execuções, proporcionando menores tempos de espera e consumo de memória para o processamento. **FUNCTIONS** oportunizam a execução de cálculos complexos, baseados nos dados de colunas, retornando tabelas. **PROCEDURES** servem para armazenar procedimentos de seleções, exclusões e mudanças de dados ou registros.