

Linguagem SQL

A linguagem SQL surgiu em meados da década de 70, sendo resultado de um estudo de E. F. Codd, membro do laboratório de pesquisa da IBM em San Jose, Califórnia. Este estudo tinha foco em desenvolver uma linguagem que adapta-se ao modelo relacional. O primeiro sistema de BD baseado em SQL tornou-se comercial no final dos anos 70 juntamente com outros sistema de BD's relacionais.

O sucesso da linguagem SQL foi tão grande que obrigou o ANSI (American National Standards Institute), a padronizar as implementações da linguagem, assim, nos dias de hoje, a maior parte de BD's seguem criteriosamente esta padronização, podendo ter algumas variações, mais mesmo assim não afetando na padronização global da linguagem tornando assim a portabilidade mais fácil, se seguida de forma adequada pelo DBA.

Em 1982, foi lançada a primeira versão padronizada da linguagem SQL, que vieram ganhando melhorias de acordo com sua evolução e tornando-se assim, a mais poderosa ferramenta para definição e manipulação de BD's e hoje utilizada em grande parte dos BD existente, tais como MySQL, SQLServer, Firebird dentre outros.

SQL: Linguagem De Múltiplo Uso

A linguagem SQL tem papel muito importante nos SGDBs atuais, podendo ter diversos enfoques:

Linguagem de Manipulação de Dados ou DML (Data Manipulation Language): A DML é um subconjunto da linguagem SQL, utilizada para Selecionar(SELECT), Inserir(INSERT), Atualizar(UPDATE) e Apagar(DELETE).

Linguagem de Definição de Dados ou DDL (Data Definition Language): A DDL permite ao usuário permite a manipulação de tabelas e elementos associados, tipo chave primária e chaves estrangeira, índices, etc. Os principais comandos são CREATE, DROP, ALTER(em algumas situações).

Linguagem de Controle de Dados ou DCL (Data Control Language): A DCL controla os aspectos destinados a autorização de dados e licenças de usuários para manipulação de dados dentro do BD. Alguns comandos comuns são GRANT(dá privilégios para usuários), REVOKE (revoga privilégios de usuários), COMMIT(em resumo grava dados no BD) e ROLLBACK(descarta dados existentes desde o último COMMIT).

Linguagem de Consultas de Dados ou DQL (Data Query Language): embora na DQL exista somente um comando(SELECT) é o mais utilizado, principalmente para consultas parametrizadas. Lembre que o SELECT também é considerado um comando DML. São exemplos os apresentados na Listagem 1.

Listagem 1. Exemplos de DQL

```
1SELECT nome FROM clientes; SELECT id,nome FROM pessoas WHERE id = :id;
```

Complementando a DQL, temos também uma série de Cláusulas (FROM, WHERE, GROUP BY, HAVING, ORDER BY, DISTINCT), Operadores Lógicos (AND, OR, NOT), Operadores de Comparação (<, >, <=, >=, =, <>, BETWEEN, LIKE) e Funções de Soma (AVG, COUNT, SUM, MIN, MAX).

SQL

SQL (Structured Query Language) é a linguagem padrão universal para manipular bancos de dados relacionais através dos SGDBs. Isso significa que todos os SGDBs (Sistema de Gerenciamento de Banco de Dados Relacionais) oferecem uma interface para acessar o banco de dados utilizando a linguagem SQL, embora com algumas variações. Logo, saber o que é SQL e como utilizá-la é fundamental para qualquer desenvolvedor de softwares.

A "Linguagem Estruturada de Consultas" (SQL, traduzida para o português) é utilizada para interagir com o SGDB e executar várias tarefas como inserir e alterar registros, criar objetos no banco de dados, gerenciar usuário, consultar informações, controlar transações, etc. Todas as operações realizadas no banco de dados podem ser solicitadas ao SGDB utilizando esta linguagem.

A linguagem SQL é dividida em 4 agrupamentos de acordo com o tipo de operação a ser executada no banco de dados. A saber, DML (Data Manipulation Language, ou Linguagem de Manipulação de Dados em português), DDL (Data Definition Language, ou Linguagem de Definição de Dados em português), DCL (Data Control Language, ou Linguagem de Controle de Dados em português) e DTL (Data Transaction Language, ou Linguagem de Transação de Dados em português).

Alguns autores classificam também uma divisão da linguagem para consultas, a DQL (Data Query Language, Linguagem de Consulta de Dados), que tem apenas um comando (SELECT), porém é mais comum encontrar este comando como integrante da DML, juntamente com os comandos INSERT, UPDATE e DELETE. Vejamos os comandos SQL de cada agrupamento.

Dml - Data Manipulation Language

DML (Linguagem de Manipulação de Dados) é o subconjunto mais utilizado da linguagem SQL, pois é através da DML que operamos sobre os dados dos bancos de dados com instruções de inserção, atualização, exclusão e consulta de informações. Os comandos SQL desse subconjunto são:

INSERT: utilizado para inserir registros (tuplas), em uma tabela.

Exemplo: `INSERT into CLIENTE(ID, NOME) values(1,'José');`

UPDATE: utilizado para alterar valores de uma ou mais linhas (tuplas) de uma tabela.

Exemplo: `UPDATE CLIENTE set NOME = 'João' WHERE ID = 1;`

DELETE: utilizado para excluir um ou mais registros (tupla) de uma tabela.

Exemplo: `DELETE FROM CLIENTE WHERE ID = 1;`

SELECT: O principal comando da SQL, o comando select é utilizado para efetuar consultas no banco de dados.

Exemplo: `SELECT ID, NOME FROM CLIENTE;`

Nota: Registro, Linha e Tupla são palavras sinônimas para referenciar a uma linha da tabela.

DDL - DATA DEFINITION LANGUAGE

DDL (Linguagem de Definição de Dados) é o subconjunto da SQL utilizado para gerenciar a estrutura do banco de dados. Com a DDL podemos criar, alterar e remover objetos (tabelas, visões, funções, etc.) no banco de dados. Os comandos deste subconjunto são:

CREATE: utilizado para criar objetos no banco de dados.

Exemplo (criar uma tabela): `CREATE TABLE CLIENTE (ID INT PRIMARY KEY, NOME VARCHAR(50));`

ALTER: utilizado para alterar a estrutura de um objeto.

Exemplo (adicionar uma coluna em uma tabela existente): `ALTER TABLE CLIENTE ADD SEXO CHAR(1);`

DROP: utilizado para remover um objeto do banco de dados.

Exemplo (remover uma tabela): `DROP TABLE CLIENTE;`

DCL - Data Control Language

DCL (Linguagem de Controle de Dados) é o subconjunto da SQL utilizado para controlar o acesso aos dados, basicamente com dois comandos que permite ou bloqueia o acesso de usuários a dados. Vejamos estes comandos:

GRANT: Autoriza um usuário a executar alguma operação.

Exemplo (dar permissão de consulta na tabela cliente para o usuário carlos): GRANT select ON cliente TO carlos;

REVOKE: Restringe ou remove a permissão de um usuário executar alguma operação.

Exemplo (não permitir que o usuário carlos crie tabelas no banco de dados): REVOKE CREATE TABLE FROM carlos;

DTL - Data Transaction Language

DTL (Linguagem de controle de transações) é o subconjunto da SQL que fornece mecanismos para controlar transações no banco de dados. São 3 comandos: iniciar uma transação (BEGIN TRANSACTION), efetivar as alterações no banco de dados (COMMIT) e cancelar as alterações (ROLLBACK).

Quem quer trabalhar com desenvolvimento de softwares precisa aprender a SQL, pois a maioria dos sistemas de informação interage com banco de dados, e essa é a linguagem universal para fazer qualquer coisa nos bancos de dados relacionais (o tipo de banco de dados mais utilizado na indústria). Pode haver pequenas variações na linguagem dependendo do SGBD, mas a sintaxe dos comandos são muito parecidas.

Cada comando citado neste artigo possui uma série de recursos, o comando que tem mais recursos, obviamente, é o comando SELECT. O objetivo deste artigo é apenas apresentar a linguagem SQL e seus comandos, continue ligado aqui no { Dicas de Programação } que vamos ver os detalhes de cada comando desta linguagem.

ANSI

A ANSI (American National Standards Institute) promove o desenvolvimento de Normas Nacionais Americanas (ANS) ao dar acreditação os procedimentos de organizações desenvolvedoras de normas (SDOs). Esses grupos cooperam para desenvolver normas voluntárias de consenso nacional. A acreditação pela ANSI significa que os procedimentos usados pelo órgão de normas em ligação ao desenvolvimento de Normas Nacionais Americanas cumprem os requisitos essenciais do instituto de abertura, equilíbrio, consenso e processo justo.

O processo de ANS é feito para resistirem análises enquanto protege os direitos e interesses de cada participante. Essencialmente, normas nacionais da ANSI aceleram a aceitação de produtos no mercado enquanto deixam claro como melhorar a segurança desses produtos para a proteção dos consumidores.

Os passos do processo de Normas Nacionais Americanas incluem:

Consenso sobre uma norma proposta por um grupo ou “órgão de consenso”, que inclui representantes de partes interessadas e substancialmente afetadas

Revisão pública em base ampla e comentário sobre minutas de normas

Consideração de e resposta a comentários enviados por associados com poder de voto do órgão de consenso relevante e por comentadores de revisão pública

Incorporação de mudanças aprovadas em uma minuta de norma

Direito de apelação para qualquer participante que ache que os princípios de processo justo não foram suficientemente respeitados durante o desenvolvimento de normas de desenvolvimento conforme os procedimentos acreditados pela ANSI do desenvolvedor de normas

Como mencionado acima, para manter a acreditação da ANSI, os desenvolvedores de normas precisam seguir consistentemente uma série de requisitos ou procedimentos que governam o processo de desenvolvimento de consenso. Esses requisitos são definidos em um documento conhecido como “Requisitos Essenciais da ANSI”; uma série de documentos de orientação ajuda a explicar melhor esses procedimentos.

Um processo justo é a chave para assegurar que ANSs sejam desenvolvidas em um ambiente equitativo, acessível e responsivo às exigências de vários colaboradores. O processo aberto e justo da ANSI

garante que todas as partes interessadas e afetadas tenham uma oportunidade de participar no desenvolvimento de uma norma. Ele também atende e protege o interesse público, já que desenvolvedores de normas com acreditação da ANSI precisam cumprir os requisitos essenciais do instituto e outras garantias de um processo justo.

[illegible]