

# LINGUAGEM SQL

*Structured Query Language*



Os bancos de dados relacionais compartilham entre si, uma característica muito interessante, todos eles usam uma linguagem chamada SQL.

SQL não é uma linguagem de programação. É uma linguagem que é usada exclusivamente para criar tabelas, manipular os dados das tabelas e principalmente, consultar os dados. É através do SQL, que os programas interagem com um banco de dados relacional.

Alguns comandos SQL mudam de um banco para outro, mas no geral, a sintaxe é a mesma entre todos os SGBDs relacionais.

### Grupos de Comandos SQL

**DDL** - Linguagem de Definição de Dados

**DML** - Linguagem de Manipulação de Dados

**DQL** - Linguagem Consulta de Dados

## DDL - Linguagem de Definição de Dados

É um conjunto de comandos SQL utilizados para definir e modificar a estrutura de objetos em um banco de dados. Esses comandos permitem criar, alterar e excluir objetos como tabelas, índices, e restrições.

### COMANDOS

CREATE = criar um novo banco de dados, tabelas...

ALTER = alterar a estrutura da tabela.

DROP = remover uma tabela, usuário...

## **DML** - Linguagem de Manipulação de Dados

É utilizado para gerenciar e manipular dados em um banco de dados. Os comandos DML permitem que os usuários insiram, atualizem e excluam dados de tabelas de banco de dados.

### **COMANDOS**

INSERT = cria um novo registro.

UPDATE = modifica registros existentes.

DELETE = exclui um ou mais registros.

## **DQL** - Linguagem de Consulta de Dados

Comandos utilizados para realizar consultas em um banco de dados.

`SELECT` = vai recuperar dados de uma tabela.

```
CREATE DATABASE biblioteca;
```

```
CREATE  
DATABASE  
biblioteca;
```



Visualizando os bancos de dados

```
SHOW DATABASES;
```

Comando para usar o banco de dados

```
USE db_biblioteca;
```

```
DROP DATABASE db_biblioteca;
```

## Constraints (Restrições)

São regras aplicadas às colunas de uma tabela para garantir a integridade e a precisão dos dados impondo regras específicas sobre esses dados.

**NOT NULL** = para a coluna não possuir valores nulos(NULL)

**UNIQUE** = garante todos os valores únicos na coluna.

**PRIMARY KEY** = identifica unicamente cada linha.

**FOREIGN KEY** = garante a integridade referencial entre tabelas.

**DEFAULT** = define um valor padrão na coluna se nenhum valor for registrado.

## Tipos de Dados

### Principais Tipos de Dados:

**INTEGER** ou **INT** = Representa números inteiros, é frequentemente usado para armazenar identificadores únicos.

**VARCHAR** = Se ajusta de acordo com o tamanho.

**CHAR** = Tamanho fixo

**DATE** = Permite armazenar datas.

**TIME** = Permite armazenar horários.

**DECIMAL** ou **DEC** = ele dará as casas decimais que você pedir.

```
CREATE TABLE tb_aluno (  
  id_aluno INT ,  
  nome VARCHAR(125),  
  email VARCHAR(125),  
  data_nascimento DATE,  
  cpf VARCHAR(100)  
);
```

