

## **CONSIDERAÇÕES IMPORTANTES**

**Este guia foi feito com base no uso da linguagem SQL no SQL SERVER**

SQL (Structured Query Language) é uma linguagem usada para se comunicar com bancos de dados relacionais. Com ela, podemos criar bancos, tabelas, inserir dados, atualizar, buscar, deletar e muito mais.

**OBS ->** O SQL pode ser case sensitivy ou não dependendo da IDE e da forma que a mesma foi instalada (este guia utiliza tudo em minúsculo).

**OBS.2->** A tabela utilizada nos exemplos é:

```
create table alunos (  
  matricula INT,  
  nome VARCHAR(100),  
  cpf VARCHAR(20),  
  rg VARCHAR(20),  
  uf_rg VARCHAR(2),  
  idade INT,  
  curso VARCHAR(100)  
);
```

Os nomes utilizados nos exemplos podem ser alterados para o contexto de uso do programador e de suas tabelas/banco de dados.

## **Categorias de Comandos SQL**

Os comandos em SQL são separados em 4 categorias principais.

**DDL ->** Data Definition Language. Cria ou altera estruturas de dados.

Exemplo: Create, alter, drop.

**DML ->** Data Manipulation Language. Manipula os dados.

Exemplo: Insert, Update, delete.

**DQL ->** Data Query Language. Consulta os dados.

Exemplo: Select.

**DCL ->** Data Control Language. Controla permições.

Exemplo: Grant, Revoke.

## COMANDOS SQL

### Comandos de Criação: DDL

#### 1) Constraints

São restrições que garantem a qualidade e consistência dos dados nas tabelas.

primary key -> Define que um campo da tabela é a chave primaria.

not null -> Impede que um campo receba valores nulos.

unique -> Garante que os valores não se repitam na tabela.

default -> Define um valor padrão para um campo.

check -> Impede valores fora de uma regra.

foreign key -> Define que um campo da tabela é uma chave estrangeira vinda de outra tabela.

```
create table Clientes{
    id int primary key,
    nome varchar(50) not null,
    cpf char(11) unique,
    idade int check (idade >= 18),
    cidadeID int,
    constraint FK_cidade foreign key (cidadeID)
    references Cidades(id)
}
```

```
Create table Cidades{
    id int primary key,
    nome varchar(100) notnull
}
```

#### 2) Create

São responsáveis pela criação de tabelas e bancos de dados.

create database Nome\_banco -> Cria um banco de dados.

Exemplo: create database banco1 -> Cria um banco de dados chamado banco1.

### Comandos de Seleção: DQL

São utilizados para consultar dados de tabelas.

\* -> O operador “\*” é um coringa utilizado para referenciar todas as colunas de uma tabela.

select \* from alunos -> Retorna todas as colunas da tabela Alunos

### **PROPRIEDADES/FUNÇÕES**