



SQL

DDL – *Data Definition Language*

Herysson R. Figueiredo
herysson.figueiredo@ufn.edu.br



Histórico

O nome **SQL** hoje é expandido como *Structured Query Language* (Linguagem de Consulta Estruturada). Originalmente, SQL era chamada de **SEQUEL** (**S**tructured **E**nglish **QUE**ry **L**anguage) e foi criada e implementada na **IBM Research** como a interface para um sistema de banco de dados relacional experimental, chamado *SYSTEM R*. A SQL agora é a linguagem padrão para SGBDs relacionais comerciais

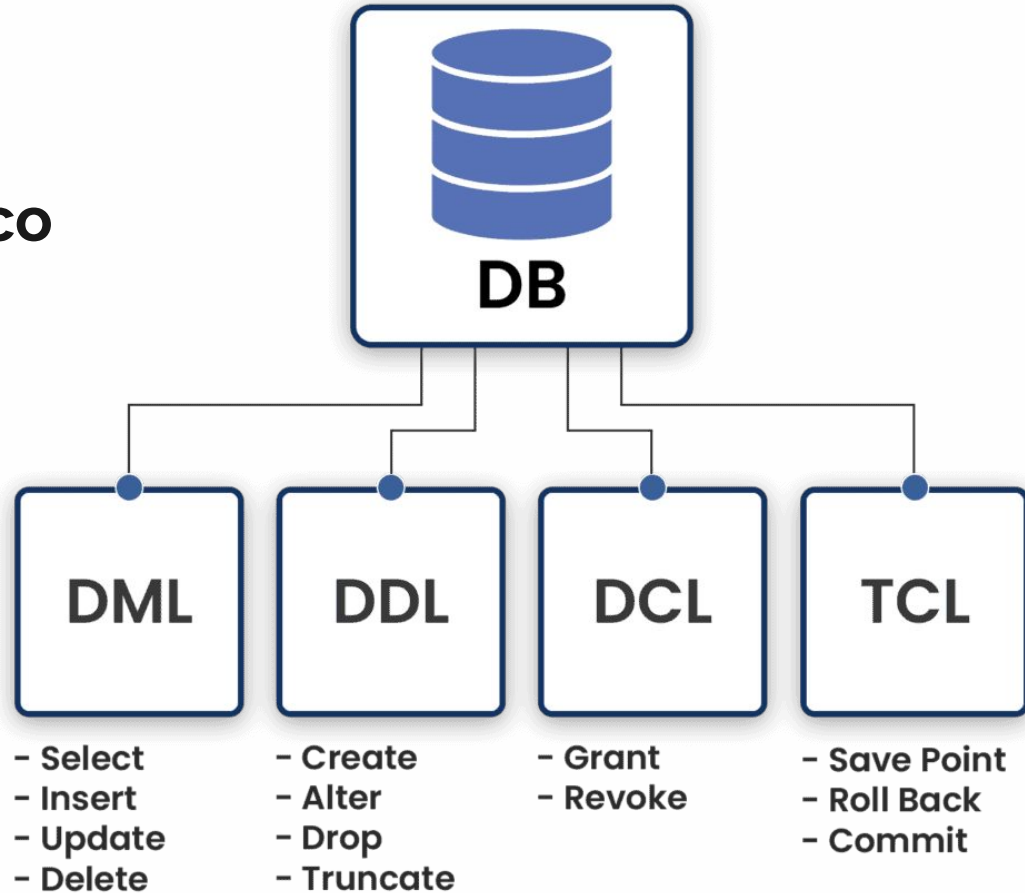


SQL Básico

A linguagem SQL é usada para interagir com bancos de dados relacionais. Com ela, é possível criar estruturas de dados, inserir, consultar, modificar, excluir informações e controlar transações e permissões.



SQL Básico





DDL – *Data Definition Language*

A **DDL** é usada para **definir e alterar a estrutura de objetos** no banco de dados, como tabelas, índices e esquemas.

Principais comandos DDL:

- **CREATE**: cria novos objetos no banco de dados (esquemas, tabelas, views, etc).
- **ALTER**: modifica a estrutura de um objeto existente.
- **DROP**: remove objetos do banco de dados.
- **TRUNCATE**: remove todos os registros de uma tabela.



DML – *Data Manipulation Language*

A **DML** é utilizada para **inserir, atualizar, deletar e consultar dados** nas tabelas.

Principais comandos DML:

- **SELECT**: consulta dados.
- **INSERT**: insere novos dados.
- **UPDATE**: altera dados existentes.
- **DELETE**: remove dados.



DCL – *Data Control Language*

A **DCL** é responsável por **controlar o acesso e permissões dos usuários** ao banco de dados.

Principais comandos DCL:

- **GRANT**: concede permissões aos usuários.
- **REVOKE**: retira permissões concedidas.



TCL – *Transaction Control Language*

A TCL controla o comportamento de transações, garantindo a integridade dos dados em operações que envolvem múltiplos comandos DML.

Principais comandos TCL:

- **BEGIN** ou **START TRANSACTION**: inicia uma transação.
- **COMMIT**: confirma as alterações feitas durante a transação.
- **ROLLBACK**: desfaz alterações realizadas durante a transação.
- **SAVEPOINT**: define um ponto dentro de uma transação para possível retorno.

Instrução CREATE



CREATE

O comando **CREATE** é uma instrução da linguagem SQL usada para **criar novos objetos** no banco de dados, como **tabelas, esquemas**, visões (views), índices, entre outros. Principais utilizações do comando CREATE:

- **CREATE DATABASE** ou **CREATE SCHEMA**
- **CREATE TABLE**



SCHEMA / DATABASE

Um **schema** é uma **estrutura de organização lógica** dentro de um banco de dados. Ele funciona como um **"container"** ou "espaço de nomes" para **armazenar objetos** como tabelas, visões, procedimentos e funções. Sua **principal função** é organizar e **agrupar objetos** relacionados, promovendo uma melhor separação de responsabilidades e maior **controle de segurança** dentro do banco de dados.



CREATE DATABASE

A instrução `CREATE DATABASE` é usada para criar um novo banco de dados SQL.

```
CREATE DATABASE nome_do_banco_de_dados;
```



Prática

Crie um banco de dados para uma biblioteca virtual.



TABLE

Uma **tabela** é a **estrutura de armazenamento de dados** em um banco de dados relacional. Ela funciona como uma grade organizada em linhas e colunas, onde cada **linha** representa um **registro** (ou tupla) e cada **coluna** representa um **atributo**.



CREATE TABLE

O comando **CREATE TABLE** é utilizado para criar uma nova tabela em um banco de dados.



CREATE TABLE

```
CREATE TABLE nome_da_tabela (  
    nome_coluna1 tipo_de_dado1 restricoes,  
    nome_coluna2 tipo_de_dado2 restricoes,  
    ...  
);
```




Tipos de Dados - *Data Types*

Cada coluna em uma tabela de banco de dados deve ter um nome e um tipo de dado. Existem três principais tipos de dados: texto, numérico e data e hora.

Cada SGBD tem o seu conjunto de tipo de dados ex:

<https://dev.mysql.com/doc/refman/8.4/en/data-types.html>

<https://learn.microsoft.com/en-us/sql/t-sql/data-types/data-types-transact-sql?view=sql-server-ver16>



Texto - *String*

Tipo	Descrição	Exemplo
CHAR(n)	Texto fixo com comprimento n	CHAR(5) → 'Ana '
VARCHAR(n)	Texto variável com comprimento até n	VARCHAR(50) → 'Maria'
TEXT	Texto longo (até 65.535 caracteres)	Descrições, observações
BLOB(tamanho)	Para BLOBs (Objetos Binários Grandes)	Imagens



Numérico - Numeric

Tipo	Descrição	Exemplo
INT	Número inteiro	100, -5
DECIMAL(m,d)	Número exato com casas decimais	DECIMAL(10,2) → 1234.56
FLOAT(p)	Número real de precisão simples	3.14
BOOLEAN	Armazena valores binários	0 ou 1 (0 → Falso 1 → Verdadeiro)



Data e Hora

Tipo	Descrição	Exemplo
DATE	Data (AAAA-MM-DD)	'2025-09-14'
DATETIME	Data e hora	'2025-09-14 12:30:45'
TIMESTAMP	Data e hora (usado para log automático)	'2025-09-14 12:30:45'
TIME	Hora (HH:MM:SS)	'14:20:00'
YEAR	Ano (formato AAAA)	'2025'



RESTRIÇÕES - CONSTRAINTS

Restrições são usadas para **limitar o tipo de dados** que podem ser inseridos em uma tabela.



RESTRIÇÕES - CONSTRAINTS

- **NOT NULL** - Garante que uma coluna não possa ter um valor NULL
- **UNIQUE** - Garante que todos os valores em uma coluna sejam diferentes
- **PRIMARY KEY** - Uma combinação de NOT NULL e UNIQUE. Identifica de forma única cada linha em uma tabela
- **FOREIGN KEY** - Impede ações que destruiriam as ligações entre tabelas
- **CHECK** - Garante que os valores em uma coluna satisfaçam uma condição específica
- **DEFAULT** - Define um valor padrão para uma coluna se nenhum valor for especificado
- **AUTO INCREMENT** - Incrementa automaticamente o valor da coluna para cada nova linha inserida, geralmente usado com chaves primárias



CHECK

A restrição **CHECK** é uma **regra aplicada a uma coluna** de uma tabela em SQL para garantir que os valores inseridos atendam a uma condição lógica específica. Ela é usada para validar os dados automaticamente no momento da inserção ou atualização.

```
valor INT CHECK (valor BETWEEN 0 AND 100);  
sexo CHAR(1) CHECK (sexo IN ('M', 'F', 'N'));
```



DEFAULT

A restrição **DEFAULT** define um **valor padrão para uma coluna** quando nenhum valor é informado durante a inserção. Ela é usada para garantir que a coluna tenha um valor inicial definido automaticamente, caso o usuário não forneça um

status **VARCHAR(10) DEFAULT 'Ativo';**

nivel_acesso **INT DEFAULT 1;**

Prática

Crie o banco representado no diagrama ER conceitual abaixo.



Instrução DROP



DROP

O comando **DROP** é usado para **remover objetos do banco de dados** de forma **definitiva**. Uma vez executado, não é possível recuperar os dados ou estrutura excluída, a não ser que exista um backup.



DROP

`DROP TABLE nome_da_tabela;`

`DROP DATABASE nome_do_banco_de_dados;`

Instrução ALTER



ALTER TABLE

O comando **ALTER TABLE** pode assumir várias formas, dependendo da operação que você deseja realizar.



ALTER TABLE - Adicionar uma Nova Coluna

ALTER TABLE nome_da_tabela

ADD nome_da_nova_coluna tipo_de_dado restricoes;



Prática

Adicionar coluna de gênero ao livro.



ALTER TABLE - Remover uma Coluna


```
ALTER TABLE nome_da_tabela
```

```
DROP COLUMN nome_da_coluna;
```



Prática

Remover a coluna genero.



ALTER TABLE - Modificar o Tipo de Dado de uma Coluna

```
ALTER TABLE nome_da_tabela
```

```
MODIFY COLUMN nome_da_coluna novo_tipo_de_dado;
```



Prática

Modificar o tipo da coluna nacionalidade, a nacionalidade será representada pela sigla do país.



ALTER TABLE - Modificar nome e tipo de uma coluna

ALTER TABLE nome_da_tabela

CHANGE nome_antigo novo_nome novo_tipo;



Prática

Vamos agora modificar o identificador da tabela Livro, o novo identificador agora deve ser o ISBN.



ALTER TABLE - Adicionar uma Restrição de Chave Estrangeira

ALTER TABLE nome_da_tabela

ADD CONSTRAINT nome_da_restricao

FOREIGN KEY (nome_da_coluna) REFERENCES

outra_tabela(nome_da_coluna_na_outra_tabela);



FOREIGN KEY - Cláusulas de ação

A restrição **FOREIGN KEY** pode ser combinada com cláusulas de ação de disparo referencial, permitindo que o projetista do esquema especifique o que deve acontecer quando o valor referenciado for excluído (**ON DELETE**) ou atualizado (**ON UPDATE**). As opções incluem:

- **SET NULL**: define o valor da chave estrangeira como NULL.
- **CASCADE**: propaga a alteração ou exclusão para a tabela dependente.
- **SET DEFAULT**: define o valor padrão da coluna (se existir).



Prática

Vamos criar uma nova tabela chamada Editora para armazenar as informações das editoras responsáveis pelos livros. Cada livro deve estar associado a uma única editora, enquanto uma editora pode ter vários livros publicados, caracterizando um relacionamento de um para muitos. A tabela deve conter dados como identificador da editora, nome, cidade, site institucional e ano de fundação.



Prática

Agora queremos garantir que o campo nacionalidade da tabela autor não seja nulo e que o campo ano_nascimento contenha valores coerentes (por exemplo, o autor deve ter nascido entre 1900 e 2025).



Prática:

Crie a data base EMPRESA representada pela imagem ao lado.

FUNCIONARIO

Pnome	Minicial	Unome	<u>Cpf</u>	Datanasc	Endereco	Sexo	Salario	Cpf_supervisor	Dnr
-------	----------	-------	------------	----------	----------	------	---------	----------------	-----

DEPARTAMENTO

Dnome	<u>Dnumero</u>	Cpf_gerente	Data_inicio_gerente
-------	----------------	-------------	---------------------

LOCALIZACAO_DEP

<u>Dnumero</u>	<u>Dlocal</u>
----------------	---------------

PROJETO

Projnome	<u>Projnumero</u>	Projlocal	Dnum
----------	-------------------	-----------	------

TRABALHA_EM

<u>Fcpf</u>	<u>Pnr</u>	Horas
-------------	------------	-------

DEPENDENTE

<u>Fcpf</u>	<u>Nome_dependente</u>	Sexo	Datanasc	Parentesco
-------------	------------------------	------	----------	------------



Referência Bibliográfica

HEUSER, Carlos Alberto. Projeto de banco de dados. 4. ed. Porto Alegre: Sagra Luzzatto, 2001. 204 p.

SILBERSCHATZ, Abraham; KORTH, Henry F.; SUDARSHAN, S. Sistema de banco de dados. Rio de Janeiro, RJ: Elsevier, 2006. 781 p.

ELMASRI, Ramez; Navathe, Shamkant B.. Sistemas de banco de dados, 7^a ed., 2018.

W3SCHOOL, MySQL Database, <https://www.w3schools.com/mysql/> acessado em 10/02/2023;