Professores
Celso Rodrigo Giusti
Daniel Manoel Filho
Marlon P. F. Rodrigues

Linguagem SQL
Comandos DML

SENAI

BANCO DE DADOS

O que é DML?

DML significa **Data Manipulation Language** (Linguagem de Manipulação de Dados). Após criarmos a estrutura do banco (DDL), usamos DML para interagir com os *dados* dentro das tabelas.

É o conjunto de comandos que nos permite:

Inserir (INSERT)
Atualizar (UPDATE)
Excluir (DELETE)
Consultar (SELECT)

BANCO DE DADOS

Comando INSERT INTO

O comando INSERT é usado para adicionar novos registros (linhas) a uma tabela.

Lembre-se: A ordem dos valores deve corresponder exatamente à ordem das colunas especificadas.

```
INSERT INTO tbl_autor (nome_autor, nacionalidade)
VALUES ('Machado de Assis', 'Brasileira');

INSERT INTO tbl_autor (nome_autor, nacionalidade)
VALUES ('J.K. Rowling', 'Britânica');
```

BANCO DE DADOS

Comando UPDATE

O comando UPDATE é usado para modificar registros (linhas) que *já existem* em uma tabela.

Importante: A cláusula WHERE é crucial. Ela especifica *quais* registros devem ser atualizados.

```sql
UPDATE tbl_autor
SET nacionalidade = 'Brasileiro'
WHERE id_autor = 1;
```

```sql
UPDATE tbl_autor
SET nome_autor = 'J.K. Rowling (Joanne Rowling)',
    nacionalidade = 'Britânica (Reino Unido)'
WHERE id_autor = 2;
```

BANCO DE DADOS

Comando DELETE

O comando DELETE é usado para remover registros (linhas) de uma tabela.

```
DELETE FROM tbl_autor
WHERE id_autor = 2;
```

Atenção: Se este autor já estivesse vinculado a um livro na tbl_autor_livro, o SGBD poderia retornar um erro de **Restrição de Chave Estrangeira (FK)**, nos impedindo de apagar o autor antes de apagar suas ligações.

BANCO DE DADOS

UPDATE e DELETE sem WHERE

O que acontece se você esquecer a cláusula WHERE?

```sql
/* NÃO EXECUTE ESTE COMANDO! */
UPDATE tbl_autor
SET nacionalidade = 'Desconhecida';
```

TODOS os autores no banco terão sua nacionalidade alterada para 'Desconhecida’.

```sql
/* NÃO EXECUTE ESTE COMANDO! */
DELETE FROM tbl_autor;
```

TODOS os registros da tabela tbl_autor serão excluídos.

Atenção: Um UPDATE ou DELETE sem WHERE afeta **todas as linhas** da tabela. Esta é uma das causas mais comuns de perda de dados em produção. Sempre verifique seu WHERE!

BANCO DE DADOS

Comando SELECT

O comando SELECT é o mais usado do SQL. Ele é utilizado para consultar e recuperar dados de uma ou mais tabelas.
É a forma como "lemos" o que está no banco.

```sql
/* Consulta todas as colunas da tabela de autores */
SELECT * FROM tbl_autor;
```

```sql
/* Consulta apenas o nome e a nacionalidade */
SELECT nome_autor, nacionalidade
FROM tbl_autor;
```

BANCO DE DADOS

SELECT com WHERE

Assim como no UPDATE e DELETE, usamos a cláusula WHERE no SELECT para filtrar os dados e trazer apenas os registros que atendem a uma condição.

```sql
/* Consulta todas as informações do autor
cujo ID é 1.
*/
SELECT * FROM tbl_autor
WHERE id_autor = 1;


/* Consulta todos os autores que são 'Brasileiros'.
(Note o uso de aspas para texto)
*/
SELECT * FROM tbl_autor
WHERE nacionalidade = 'Brasileiro';
```

BANCO DE DADOS

RESUMO DA AULA

SENAI

✅ DML (Data Manipulation Language) = Subconjunto do SQL usado para manipular os dados (não a estrutura).

✅ INSERT INTO = Adiciona novas linhas (registros) a uma tabela.

✅ UPDATE = Modifica dados em linhas que já existem em uma tabela.

✅ DELETE = Remove linhas (registros) de uma tabela.

✅ SELECT = Consulta e recupera dados (lê os registros) de uma tabela.

✅ Cláusula WHERE = Filtra quais linhas serão afetadas pelo SELECT, UPDATE ou DELETE. É obrigatória para segurança em UPDATE e DELETE.

ATIVIDADE

Execício DML

1

Com base no esquema do banco db_saber_e_cia, execute os seguintes comandos DML. O objetivo é popular (preencher) nosso banco de dados da biblioteca.

Cadastrando Membros - Insira 3 novos membros na tbl_membro. Use dados fictícios.

(Ex: ID 101, 'Ana Silva', 'Rua A, 123', '11-98765-4321')
(Ex: ID 102, 'Bruno Costa', 'Av. B, 456', '11-91234-5678')
(Ex: ID 103, 'Carla Dias', 'Praça C, 789', '11-95555-4444’)

Consultando Membros - Execute um SELECT para verificar se os 3 membros foram inseridos corretamente na tbl_membro.

Cadastrando Livros - Insira 2 livros na tbl_livro.

(Ex: ISBN '978-85-325-3078-3', 'Harry Potter e a Pedra Filosofal', 1997, 'Rocco')
(Ex: ISBN '978-85-7126-061-0', 'Dom Casmurro', 1899, 'Editora Clássica')

ATIVIDADE

Execício DML

2

Corrigindo um Erro

O ano de publicação de 'Dom Casmurro' está incorreto no nosso registro. O SGBD aceitou 1899 (tipo YEAR), mas o correto para *esta edição* é 2019.

Execute um UPDATE na tbl_livro para corrigir o ano_publicacao do livro com o ISBN '978-85-7126-061-0' para 2019.

Consultando Livros Antigos

Execute um SELECT para buscar na tbl_livro todos os livros (neste caso, apenas 1) que foram publicados *antes* do ano 2000. Use a cláusula WHERE com o operador <.

Removendo um Membro

O membro 'Bruno Costa' (ID 102) pediu para cancelar seu cadastro.

Execute um DELETE na tbl_membro para remover o registro onde o id_membro é 102.

(Verifique com um SELECT se ele foi removido).

BANCO DE DADOS

Refinando Nossas Consultas

Aprendemos o DML (INSERT, UPDATE, DELETE, SELECT) e o filtro básico com WHERE.

Vamos "turbinar" nossas consultas.

- **Operadores:** Permitem criar filtros mais complexos e inteligentes (ex: buscar por faixas de data, por nomes que *começam com 'A'*, ou que *não são* de uma categoria).
- **Funções:** São ferramentas prontas do SQL que executam cálculos, manipulam textos ou datas (ex: formatar um nome para maiúsculas, calcular a data de devolução de um livro).

BANCO DE DADOS

Operadores Aritméticos

Usados para realizar cálculos matemáticos, geralmente dentro de um SELECT.

+ (Adição) - (Subtração) * (Multiplicação) / (Divisão)

```
SELECT titulo_livro, ano_publicacao, ano_publicacao + 10 AS ano_revisao
FROM tbl_livro;
```

BANCO DE DADOS

Operadores Relacionais

Usados (principalmente no WHERE) para comparar valores. Já usamos o = anteriormente.

- = (Igual)
- != ou <> (Diferente)
- > (Maior que)
- < (Menor que)
- >= (Maior ou igual a)
- <= (Menor ou igual a)

```sql
/* Seleciona livros publicados ANTES do ano 2000 */
SELECT * FROM tbl_livro
WHERE ano_publicacao < 2000;
```

BANCO DE DADOS

Operadores Lógicos (AND, OR)

Usados no WHERE para combinar múltiplas condições.

AND: (E) Todas as condições devem ser verdadeiras.
OR: (OU) Pelo menos uma das condições deve ser verdadeira.

```sql
/* Busca livros da 'Rocco' E que foram publicados
   antes de 2010.
*/
SELECT * FROM tbl_livro
WHERE editora = 'Rocco'
  AND ano_publicacao < 2010;
```

```sql
/* Busca membros que se chamam 'Ana' OU moram
   em um endereço que contém 'Rua A'.
*/
SELECT * FROM tbl_membro
WHERE nome_membro = 'Ana Silva'
   OR endereco = 'Rua A, 123';
```

BANCO DE DADOS

Operadores Lógicos (NOT)

O operador NOT é usado para negar uma condição, ou seja, buscar o oposto.

```
/* Busca todos os autores que NÃO são 'Brasileiros' */
SELECT * FROM tbl_autor
WHERE NOT nacionalidade = 'Brasileiro';
```

É o mesmo que usar o operador relacional "diferente" (!=)

```
SELECT * FROM tbl_autor
WHERE nacionalidade != 'Brasileiro';
```

BANCO DE DADOS

Operadores Auxiliares (BETWEEN, IN)

São atalhos para filtros comuns:

BETWEEN: (ENTRE) Seleciona valores dentro de um intervalo (inclusive).
IN: (EM) Seleciona valores que correspondem a uma lista.

```sql
/* Livros publicados entre 1990 e 2000 (inclusive) */
SELECT * FROM tbl_livro
WHERE ano_publicacao BETWEEN 1990 AND 2000;

/* É o mesmo que:
   WHERE ano_publicacao >= 1990
     AND ano_publicacao <= 2000
*/
```

```sql
/* Busca livros que são da editora 'Rocco' OU 'Editora
   Clássica'
*/
SELECT * FROM tbl_livro
WHERE editora IN ('Rocco', 'Editora Clássica');
```

BANCO DE DADOS

Operador Auxiliar (LIKE)

O LIKE é usado para busca de padrões em colunas de texto (VARCHAR). É a base de qualquer "campo de busca".

Usa caracteres curinga (wildcards):

% (Porcento): Substitui zero, um ou múltiplos caracteres.

_ (Underline): Substitui *exatamente um* caractere.

```sql
/* Busca membros cujo nome COMEÇA com 'Ana' */
SELECT * FROM tbl_membro
WHERE nome_membro LIKE 'Ana%';

/* Busca livros que CONTÊM 'Potter' no título */
SELECT * FROM tbl_livro
WHERE titulo_livro LIKE '%Potter%';
```

BANCO DE DADOS

Operador Auxiliar (IS NULL)

Um valor NULL é diferente de 0 ou de '' (texto vazio). NULL significa "ausência de valor" ou "dado desconhecido".

Não podemos usar = NULL. Devemos usar IS NULL ou IS NOT NULL.

```
/* Busca todos os empréstimos que AINDA NÃO FORAM
   DEVOLVIDOS (onde a data de devolução efetiva
   está nula)
*/
SELECT * FROM tbl_emprestimo
WHERE data_devolucao_efetiva IS NULL;
```

BANCO DE DADOS

Funções SQL

Funções são "caixas de ferramentas" prontas do SQL. Elas recebem um valor (ou uma coluna) e retornam um novo valor modificado.

Vamos ver 4 categorias:

- Funções de Data e Hora
- Funções de String (Texto)
- Funções Matemáticas
- Funções de Agregação

BANCO DE DADOS

Funções de Data e Hora

Manipulam e extraem informações de colunas de data/hora.

- NOW(): Retorna a data E hora atuais do servidor.
- CURDATE(): Retorna apenas a DATA atual do servidor.
- YEAR(data): Extrai o ano de uma data.
- MONTH(data): Extrai o mês de uma data.
- DATEDIFF(data1, data2): Retorna a diferença em dias entre duas datas.

```
/* Usando em um INSERT (ex: um novo empréstimo)
   (data_devolucao é 7 dias a partir de hoje)
*/
INSERT INTO tbl_emprestimo (...)
VALUES (..., CURDATE(), CURDATE() + INTERVAL 7 DAY, ...);
```

BANCO DE DADOS

Funções de String (Texto)

Manipulam colunas de texto (CHAR, VARCHAR).

CONCAT(str1, str2, ...): Junta (concatena) duas ou mais strings.
UPPER(str): Converte todo o texto para MAIÚSCULAS.
LOWER(str): Converte todo o texto para minúsculas.
SUBSTRING(str, inicio, tamanho): Extrai um pedaço do texto.

```sql
/* Cria uma "etiqueta" de autor formatada */
SELECT CONCAT(UPPER(nome_autor), ' (', nacionalidade, ')')
       AS etiqueta
FROM tbl_autor;

/* Resultado:
   MACHADO DE ASSIS (Brasileiro)
*/
```

BANCO DE DADOS

Funções Matemáticas

Realizam cálculos matemáticos mais complexos.

ROUND(numero, casas_decimais): Arredonda um número.

CEIL(numero): Arredonda para o *próximo* número inteiro (teto).

FLOOR(numero): Arredonda para o número inteiro *anterior* (piso).

RAND(): Retorna um número aleatório.

```
/* Se tivéssemos uma tabela 'multas' com valor
   de 19.99, poderíamos calcular um reajuste de 5%
   e arredondar para 2 casas decimais.
*/
SELECT ROUND(19.99 * 1.05, 2);

/* Retorna: 20.99 */
```

BANCO DE DADOS

Funções de Agregação

Importante: Diferente das outras, funções de agregação operam em um *conjunto* de linhas (a tabela inteira, ou grupos) e retornam um **único valor** como resultado.

- COUNT(): Conta o número de linhas.
- SUM(): Soma os valores de uma coluna.
- AVG(): Calcula a média dos valores de uma coluna.
- MIN(): Encontra o menor valor em uma coluna.
- MAX(): Encontra o maior valor em uma coluna.

BANCO DE DADOS

Funções de Agregação (COUNT)

O COUNT é usado para contar registros.

COUNT(*): Conta todas as linhas da tabela. COUNT(coluna): Conta todas as linhas onde aquela coluna **NÃO É NULA**.

```
/* Quantos membros temos cadastrados no total? */
SELECT COUNT(*) AS total_membros
FROM tbl_membro;

/* Quantos empréstimos já foram devolvidos?
   (Conta apenas os que têm data de devolução)
*/
SELECT COUNT(data_devolucao_efetiva) AS total_devolvidos
FROM tbl_emprestimo;
```

BANCO DE DADOS

Funções de Agregação (MIN, MAX, AVG, SUM)

Usadas para encontrar valores específicos em colunas numéricas.

- MIN(): O menor valor.
- MAX(): O maior valor.
- AVG(): A média.
- SUM(): A soma.

```sql
/* Qual o ano de publicação do livro MAIS ANTIGO
   do nosso acervo?
*/
SELECT MIN(ano_publicacao) AS livro_mais_antigo
FROM tbl_livro;

/* E o ano do MAIS NOVO? */
SELECT MAX(ano_publicacao) AS livro_mais_novo
FROM tbl_livro;
```

BANCO DE DADOS

RESUMO DA AULA

✅ Operadores Aritméticos = Usados para cálculos matemáticos (+, -, *, /).

✅ Operadores Relacionais = Usados para comparações no WHERE (=, !=, >, <).

✅ Operadores Lógicos = Usados para combinar filtros (AND, OR, NOT).

✅ Operadores Auxiliares = Atalhos para filtros complexos (BETWEEN para faixas, IN para listas, LIKE para busca de texto, IS NULL para valores vazios).

✅ Funções (Data, String, Math) = Ferramentas que transformam dados de uma linha (ex: UPPER(), CONCAT(), CURDATE()).

✅ Funções de Agregação = Ferramentas que resumem várias linhas em um resultado (ex: COUNT(), MIN(), MAX()).

ATIVIDADE

Execício DML

3

Para os exercícios de hoje fazerem sentido, precisamos de mais dados. Execute os comandos abaixo para popular melhor o banco (revisão de INSERT).

```
INSERT INTO tbl_autor (nome_autor, nacionalidade)
VALUES ('Clarice Lispector', 'Brasileira'),
    ('George Orwell', 'Britânico'),
    ('Isaac Asimov', 'Russo-Americano’);

INSERT INTO tbl_livro (isbn, titulo_livro, ano_publicacao, editora)
VALUES ('978-85-325-2306-8', 'A Revolução dos Bichos', 1945, 'Companhia das Letras'),
    ('978-0-00-711711-0', '1984', 1949, 'Penguin Books'),
    ('978-85-325-1997-9', 'Eu, Robô', 1950, 'Aleph');
```

ATIVIDADE

Execício Operadores

3

Operador LIKE

Selecione todos os membros da tbl_membro cujo nome termina com 'Silva'. (Dica: LIKE '%Silva').

Operador BETWEEN

Selecione todos os livros da tbl_livro que foram publicados durante a Segunda Guerra Mundial (entre 1939 e 1945).

Operador IN e NOT

Selecione todos os livros da tbl_livro cuja editora seja 'Rocco' ou 'Aleph'. (Use o IN).

***Desafio**:* Altere a consulta para trazer todos, exceto os da 'Rocco' ou 'Aleph'. (Use o NOT IN).

ATIVIDADE

Exercícios (Funções)

4

Funções de String (CONCAT, UPPER)

Crie uma consulta na tbl_membro que retorne uma única coluna chamada "CONTATO" no formato: "NOME DO MEMBRO (EM MAIÚSCULO) - TELEFONE".
(Ex: "ANA SILVA - 11-98765-4321")

Funções de Agregação (COUNT, MIN, MAX)

Consulta 1: Quantos autores *brasileiros* temos na tbl_autor? (Dica: combine COUNT com WHERE).

Consulta 2: Qual é o livro mais antigo (menor ano_publicacao) da editora 'Aleph'? (Dica: combine MIN com WHERE).

Funções de Data (CURDATE)

Vamos registrar um empréstimo. O membro 101 (Ana Silva) pegou o livro 'Eu, Robô' (ISBN '978-85-325-1997-9'). O exemplar é o 1.

Insira um registro na tbl_emprestimo.

id_emprestimo: (ex: 501), **data_emprestimo**: Use a função CURDATE(),
data_devolucao: Use CURDATE() + INTERVAL 14 DAY (14 dias de prazo)
data_devolucao_efetiva: Deixe NULL, **id_exemplar**: (ex: 1 - supondo que exista)
id_membro: (ex: 101)

BANCO DE DADOS

Referências

LIVROS

DATE, C. J. **Introdução a sistemas de bancos de dados.** 8. ed. Rio de Janeiro: Elsevier, 2004.

ELMASRI, R.; NAVATHE, S. B. **Sistemas de banco de dados.** 6. ed. São Paulo: Pearson, 2011.

SENAI. DN. **Linguagem SQL.** Brasília: SENAI/DN, 2014. (Série Informática).

IA

Parte do conteúdo desta apresentação foi elaborada com o auxílio de inteligência artificial, visando complementar e organizar as informações de forma didática.

Escola SENAI "Italo Bologna"

Av. Goiás, 139 – Itu/SP

Telefone
(11) 2396-1999

Instagram
@senai.itu

Facebook
/senai.itu

Site
https://sp.senai.br/unidade/itu/