

BANCO DE DADOS RELACIONAL

Criação de banco de dados
Manipulação de Dados (INSERT, UPDATE, DELETE)

Professora:

Lucineide N. Pimenta

Objetivos da aula



✓ **Objetivos Gerais:**

- ✓ Ensinar como **inserir, modificar e excluir dados** no PostgreSQL.
- ✓ Explicar o funcionamento dos comandos **INSERT, UPDATE e DELETE**.
- ✓ Relacionar a manipulação de dados ao **desafio da ABP** para aplicação prática.

✓ **Objetivos Específicos:**

- ✓ Inserir registros no banco de dados usando **INSERT INTO**.
- ✓ Modificar dados existentes com **UPDATE**.
- ✓ Remover informações desnecessárias com **DELETE**.
- ✓ Garantir a integridade dos dados ao manipular informações.

O que é Manipulação de Dados?

❑ Definição:

- Manipulação de dados significa **adicionar, atualizar e excluir informações** armazenadas em um banco de dados.
- Esses comandos fazem parte da **Linguagem de Manipulação de Dados (DML - Data Manipulation Language)** no SQL.

❑ Por que isso é importante?

- ❑ Permite que os sistemas **atualizem informações em tempo real**.
- ❑ Evita que os dados fiquem **desatualizados ou incorretos**.
- ❑ Melhora a eficiência da aplicação e mantém o banco **organizado**.

❑ Exemplo prático:

- Um sistema de cadastro de clientes precisa **adicionar um novo cliente**,
- Depois, precisa **atualizar o telefone de um cliente**.
- Se um cliente quiser sair do sistema, a empresa pode **excluir os dados** dele.

Criando um banco

- ❑ **Criar o banco:**

CREATE DATABASE *db_escola*;

- ❑ **Conectar no banco:**

pelo pgAdmin → conectar no novo banco.

Comando INSERT (Inserindo Dados)

Exemplo prático:

No banco de dados *db_escola*, crie as tabelas *alunos* e *cursos* para inserir os dados dos alunos:

```
CREATE TABLE cursos (  
  id_curso SERIAL PRIMARY KEY,  
  nome VARCHAR(100) NOT NULL  
);  
  
CREATE TABLE alunos (  
  id_aluno SERIAL PRIMARY KEY,  
  nome VARCHAR(100) NOT NULL,  
  idade INT,  
  id_curso INT REFERENCES cursos(id_curso)  
);
```

Comando INSERT (Inserindo Dados)

- ❑ **Definição:**
O comando INSERT INTO é usado para **adicionar novos registros** em uma tabela.
- ❑ **Sintaxe básica:**
INSERT INTO nome_da_tabela (coluna1, coluna2, coluna3) VALUES (valor1, valor2, valor3);

Exemplo prático:

INSERT INTO cursos (nome) VALUES (Engenharia');

INSERT INTO alunos (nome, idade, curso) VALUES ('João Silva', 22, 'Engenharia');

Comando INSERT (Inserindo Dados)

- ❑ **Inserindo um registro:**

```
INSERT INTO cursos (nome)  
VALUES (Engenharia');
```

```
INSERT INTO alunos (nome, idade, curso)  
VALUES ('João Silva', 22, 1);
```

Agora João foi adicionado à tabela de alunos!

Comando INSERT (Inserindo Dados)

- ❑ **Inserindo múltiplos registros:**

```
INSERT INTO cursos (nome) VALUES  
(‘Análise de Sistemas’),  
(‘Computação’),  
(‘Matemática’);
```

```
INSERT INTO alunos (nome, idade, curso) VALUES  
(‘Maria Souza’, 20, 3),  
(‘Carlos Lima’, 25, 4);
```

- ❑ Com apenas **um comando**, adicionamos **vários registros**!

Comando UPDATE (Atualizando Dados)

- ❑ **Definição:**

O comando modifica informações já existentes no banco de dados.

- ❑ **Sintaxe básica:**

```
UPDATE nome_da_tabela  
SET coluna1 = novo_valor  
WHERE condição;
```

- ❑ **Exemplo prático:**

```
UPDATE alunos  
SET idade = 23  
WHERE nome = 'João Silva';
```

- ❑ Agora João tem **23 anos**, pois atualizamos a idade dele!

Comando UPDATE (Atualizando Dados)

- ❑ Atualizando múltiplos campos ao mesmo tempo:
UPDATE alunos
SET idade = 21, curso = 1
WHERE nome = 'Maria Souza';
- ❑ **Importante:** Sempre use WHERE para **evitar atualizar todos os registros!**

Comando DELETE (Excluindo Dados)

- ❑ **Definição:**
O comando remove um ou mais registros de uma tabela.
- ❑ **Sintaxe básica:**
DELETE FROM nome_da_tabela
WHERE condição;
- ❑ **Exemplo prático:**
DELETE FROM alunos
WHERE nome = 'Carlos Lima';
- ❑ Carlos foi removido da tabela **alunos**.
- ❑ Removendo todos os registros de uma tabela:
DELETE FROM alunos;
- ❑ ⚠ **Cuidado!** Esse comando **deleta todos os dados** da tabela!

Diferença entre DELETE e TRUNCATE

- ❑ **DELETE**

- ❑ Permite excluir registros específicos.
- ❑ Pode ser desfeito com **ROLLBACK** (se houver transação ativa).

- ❑ **TRUNCATE**

- ❑ Remove **todos os registros** de uma tabela de uma vez.
- ❑ **Mais rápido**, mas **não pode ser desfeito**.
- ❑ **Exemplo:**

TRUNCATE TABLE alunos;

- ❑ ⚠ Use com cuidado, pois todos os dados da tabela serão apagados!



BANCO DE DADOS RELACIONAL

Projeto ABP

Tema do Projeto ABP (Provisório)

Aplicativo Móvel de Monitoramento e Comunicação de Eventos Climáticos e Ambientais Críticos para a População.

O aplicativo será desenvolvido para o **INPE**, com foco em alertas de queimadas, inundações, desmatamento, mudanças climáticas e coleta de dados locais da população em tempo real.

Projeto ABP (clima_alerta)

□ Projeto ABP com exemplos concretos

Evento

idEvento (PK)

titulo → "Queimada em área de preservação"

descricao → "Fogo se alastrando na mata próxima à represa."

dataHora → 2025-08-15 14:35:00

status → "Ativo" (*ex.: Ativo, Em Monitoramento, Resolvido*)

idTipoEvento (FK) → 1 (Queimada)

idLocalizacao (FK) → 5 (Localização da represa)

TipoEvento

idTipoEvento (PK)

nome → "Queimada"

descricao → "Incêndio de grandes proporções em áreas urbanas ou rurais."

Localizacao

idLocalizacao (PK)

latitude → -23.305

longitude → -45.965

cidade → "Jacareí"

estado → "SP"

Usuario

idUsuario (PK)

nome → "Maria Oliveira"

email → "maria.oliveira@email.com"

senhaHash → "2b6c7f64f76b09d0a7b9e..." (hash da senha, não a senha em si)

Relato

idRelato (PK)

texto → "Fumaça intensa e chammas visíveis a partir da rodovia."

dataHora → 2025-08-15 15:10:00

idEvento (FK) → 1 (Queimada em área de preservação)

idUsuario (FK) → 2 (Maria Oliveira)

Alerta

idAlerta (PK)

mensagem → "Evacuação imediata da área próxima à represa."

dataHora → 2025-08-15 15:20:00

nivel → "Crítico" (*Baixo, Médio, Alto, Crítico*)

idEvento (FK) → 1 (Queimada em área de preservação)

Projeto ABP (clima_alerta)

□ MER corrigido (descrição textual):

•Evento

idEvento (PK)
titulo
descricao
dataHora
status
idTipoEvento (FK)
idLocalizacao (FK)

•TipoEvento

idTipoEvento (PK)
nome
Descricao

•Localizacao

idLocalizacao (PK)
latitude
longitude
cidade
estado

• Usuario

idUsuario (PK)
nome
email
senhaHash

• Relato

idRelato (PK)
texto
dataHora
idEvento (FK)
idUsuario (FK)

• Alerta

idAlerta (PK)
mensagem
dataHora
nivel
idEvento (FK)

□ Relacionamentos e cardinalidades:

•**Evento–TipoEvento**: N:1 (vários eventos podem ser do mesmo tipo).

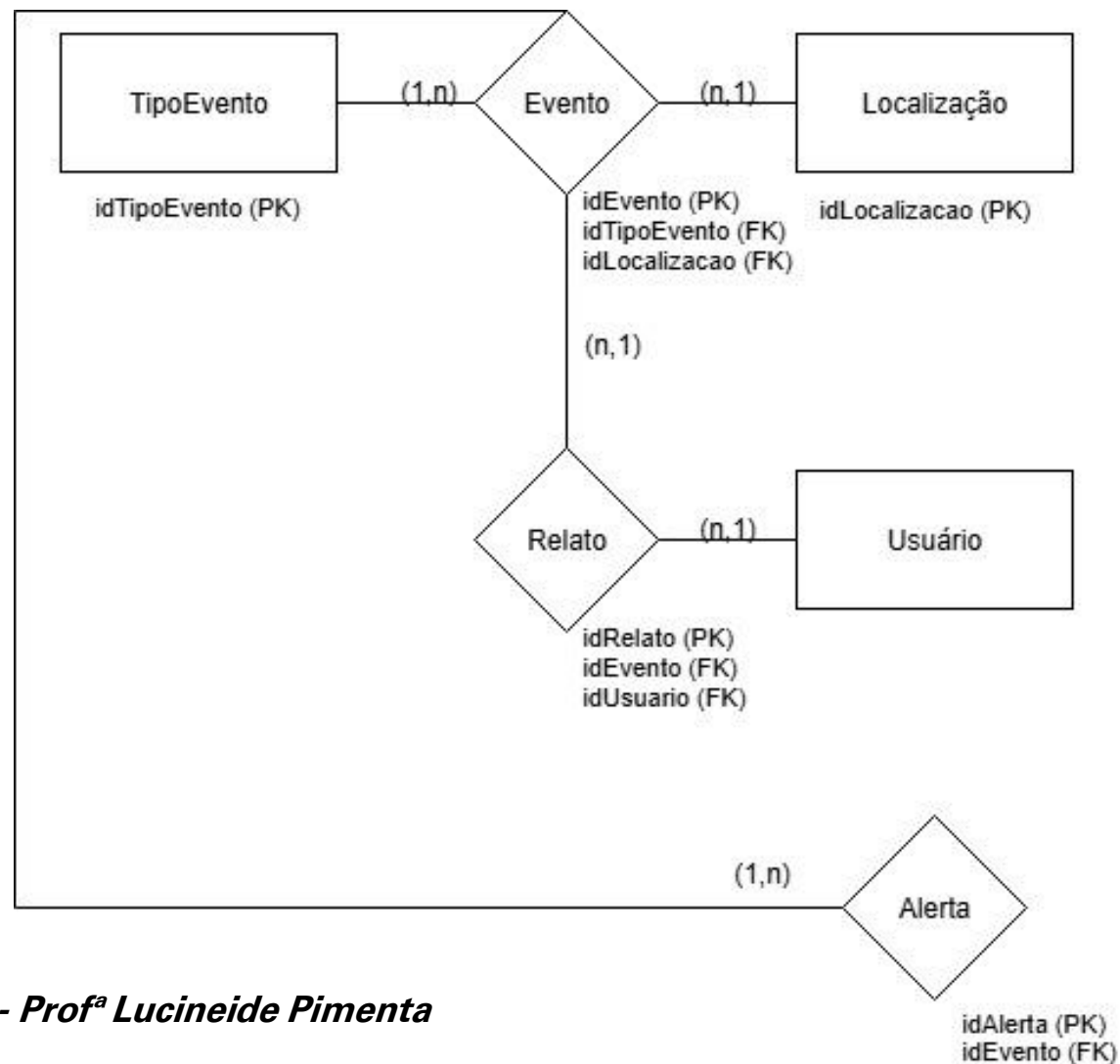
•**Evento–Localizacao**: N:1 (vários eventos podem ocorrer na mesma localização).

•**Relato–Evento**: N:1 (vários relatos podem estar vinculados a um mesmo evento).

•**Relato–Usuario**: N:1 (um usuário pode criar vários relatos).

•**Alerta–Evento**: N:1 (um evento pode ter vários alertas).

Projeto ABP (clima_alerta)





BANCO DE DADOS RELACIONAL

Exercícios Práticos

Atividade Prática (Individual)

- ❑ Criar o banco de dados *clima_alerta*.
- ❑ Criar todas as tabelas do modelo normalizado do projeto disponível na aula anterior.
- ❑ Crie uma tabela auxiliar que **não estava no modelo inicial** mas pode ser útil
 - ❑ (ex.: categoria_usuario, historico_evento).
- ❑ **Entrega:**
 - Banco criado (**clima_alerta**).
 - Script **schema.sql** com todas as tabelas.
 - Entrega no GitHub na pasta **BDR-Aula04**.

Referências Bibliográfica da Aula

Livros:

Elmasri & Navathe (2010). Sistemas de Banco de Dados.

Silberschatz et al. (2011). Sistemas de Banco de Dados.

Links úteis:

 [PostgreSQL Docs](#)

 [DBDiagram.io](#)

Bibliografia Básica

- ❑ DATE, C. J. **Introdução a sistemas de bancos de dados**. Rio de Janeiro, Elsevier: Campus, 2004.
- ❑ ELMASRI, R.; NAVATHE, S. B. **Sistemas de Banco de Dados**. 7 ed. São Paulo: Pearson, 2018.
- ❑ SILBERSCHATZ, A.; SUNDARSHAN, S.; KORTH, H. F. **Sistema de banco de dados**. Rio de Janeiro: Elsevier Brasil, 2016.

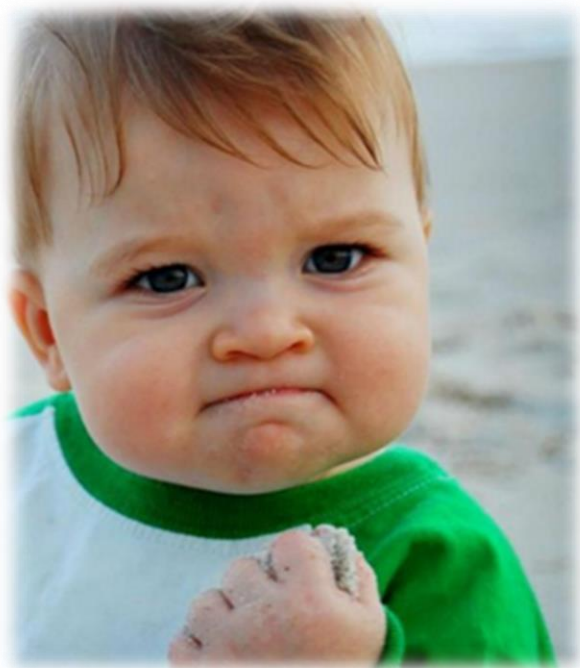
Bibliografia Complementar

- ❑ BEAULIEU, A. **Aprendendo SQL**. São Paulo: Novatec, 2010.
- ❑ GILLENSON, M. L. **Fundamentos de Sistemas de Gerência de Banco de Dados**. Rio de Janeiro: LTC, 2006.
- ❑ MACHADO, F. N. R. **Banco de Dados: Projeto e Implementação**. São Paulo: Érica, 2005.
- ❑ OTEY, M; OTEY, D. **Microsoft SQL Server 2005: Guia do Desenvolvedor**. Rio de Janeiro: Ciência Moderna, 2007.
- ❑ RAMAKRISHNAN, R.; GEHRKE, J. **Sistemas de Gerenciamento de Bancos de Dados**. 3 ed. Porto Alegre: Bookman, 2008.
- ❑ ROB, P; CORONEL, C. **Sistemas de Banco de Dados: Projeto, Implementação e Gerenciamento**. 8 ed. São Paulo: Cengage Learning, 2011.
- ❑ TEOREY, T; LIGHTSTONE, S; NADEAU, T. **Projeto e Modelagem de Bancos de Dados**. São Paulo: Campus, 2006.

Dúvidas?



Considerações Finais



**Professor(a):
Lucineide Pimenta**

Bom descanso à todos!

