Centro Paula Souza

Fatec Jacareí
Prof. Francisco de Moura

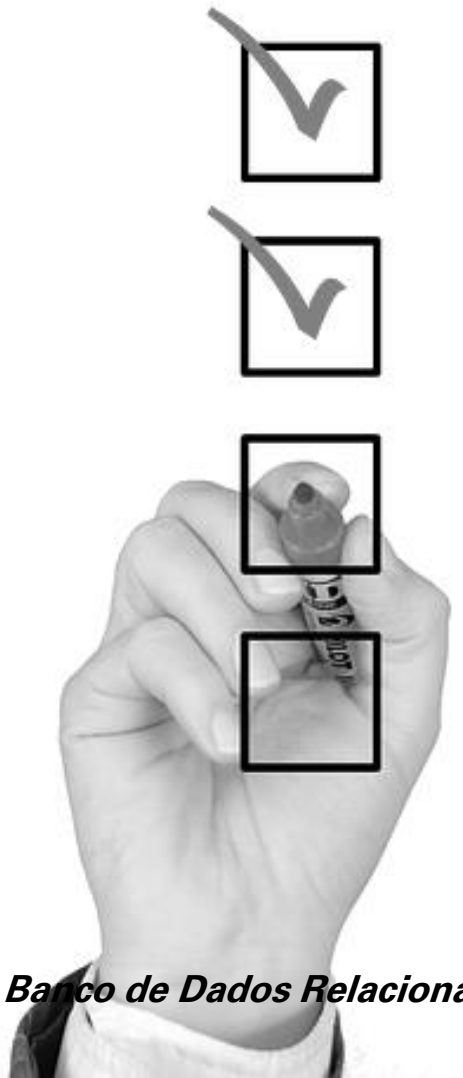
BANCO DE DADOS RELACIONAL

Consultas Básicas (SELECT, WHERE, ORDER BY)

Professora: Lucineide Pimenta

Recapilando a aula anterior

Fatec
Jacareí
Prof. Francisco de Moura



- **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.

Tema do Projeto ABP (Provisório)

Fatec
Jacareí
Prof. Francisco de Moura

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)

Fatec Jacareí
Prof. Francisco de Moura

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 chamas 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)

Fatec
Jacareí
Prof. Francisco de Moura

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).

Recapitulando:
Atividade Prática (Individual)

Fatec
Jacareí
Prof. Francisco de Moura

- 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**.

Criando o Banco de Dados

Fatec
Jacareí
Prof. Francisco de Moura

- **Criar o banco**
  - Se for no pgAdmin: botão direito em **Databases** → **Create Database**.
  - Se for pelo terminal: *CREATE DATABASE clima_alerta*;
- **Conectar no banco**
  - Se for pelo pgAdmin → conectar no novo banco.
  - Se for pelo terminal: \c *clima_alerta*;

Criando o Banco de Dados

Fatec
Jacareí
Prof. Francisco de Moura

- Agora vamos **traduzir o modelo lógico normalizado** em SQL.
  Exemplo com as tabelas principais:

```
-- Tabela de Tipos de Evento
CREATE TABLE tipo_evento (
    id_tipo_evento SERIAL PRIMARY KEY,
    nome VARCHAR(100) NOT NULL,
    descricao TEXT
);
```

```
-- Tabela de Estados
CREATE TABLE estado (
    sigla_estado CHAR(2) PRIMARY KEY,
    nome_estado VARCHAR(100) NOT NULL
);
```

Criando o Banco de Dados

Fatec
Jacareí
Prof. Francisco de Moura

- Agora vamos **traduzir o modelo lógico normalizado** em SQL.
  Exemplo com as tabelas principais:

```
-- Tabela de Localização
CREATE TABLE localizacao (
    id_localizacao SERIAL PRIMARY KEY,
    latitude NUMERIC(9,6) NOT NULL,
    longitude NUMERIC(9,6) NOT NULL,
    cidade VARCHAR(100) NOT NULL,
    sigla_estado CHAR(2) NOT NULL REFERENCES
estado(sigla_estado)
);
```

Chave Primária (PRIMARY KEY)
•Garante que cada registro é **único** na tabela.
•Já usamos com **id_localizacao SERIAL PRIMARY KEY**.

Chave Estrangeira (FOREIGN KEY)
•Conecta tabelas e garante integridade referencial.

Criando o Banco de Dados

Fatec
Jacareí
Prof. Francisco de Moura

- Agora vamos **traduzir o modelo lógico normalizado** em SQL.
  Exemplo com as tabelas principais:

```
-- Tabela de Usuários
CREATE TABLE usuario (
    id_usuario SERIAL PRIMARY KEY,
    nome VARCHAR(150) NOT NULL,
    email VARCHAR(150) UNIQUE NOT NULL,
    senha_hash VARCHAR(255) NOT NULL
);
-- Tabela de Telefones do Usuário
CREATE TABLE telefone (
    id_telefone SERIAL PRIMARY KEY,
    numero VARCHAR(20) NOT NULL,
    id_usuario INT NOT NULL REFERENCES usuario(id_usuario)
);
```

Restrição UNIQUE

- Garante que não existam valores repetidos em uma coluna.
- Exemplo: **e-mail do usuário deve ser único**.

Criando o Banco de Dados

Fatec
Jacareí
Prof. Francisco de Moura

- Agora vamos **traduzir o modelo lógico normalizado** em SQL.
  Exemplo com as tabelas principais:

```
-- Tabela de Eventos
CREATE TABLE evento (
    id_evento SERIAL PRIMARY KEY,
    titulo VARCHAR(150) NOT NULL,
    descricao TEXT,
    data_hora TIMESTAMP NOT NULL,
    status VARCHAR(30) CHECK (status IN ('Ativo','Em Monitoramento','Resolvido')),
    id_tipo_evento INT NOT NULL REFERENCES tipo_evento(id_tipo_evento),
    id_localizacao INT NOT NULL REFERENCES localizacao(id_localizacao)
);
```

Restrição CHECK

- Define condições que os dados precisam respeitar.
- **Exemplo:** status do evento deve estar dentro de uma lista de opções válidas.

Criando o Banco de Dados

Fatec
Jacareí
Prof. Francisco de Moura

- Agora vamos **traduzir o modelo lógico normalizado** em SQL.
  Exemplo com as tabelas principais:

```
-- Tabela de Relatos
CREATE TABLE relato (
    id_relato SERIAL PRIMARY KEY,
    texto TEXT NOT NULL,
    data_hora TIMESTAMP NOT NULL,
    id_evento INT NOT NULL REFERENCES
evento(id_evento),
    id_usuario INT NOT NULL REFERENCES
usuario(id_usuario)
);
```

```
-- Tabela de Alertas
CREATE TABLE alerta (
    id_alerta SERIAL PRIMARY KEY,
    mensagem TEXT NOT NULL,
    data_hora TIMESTAMP NOT NULL,
    nivel VARCHAR(20) CHECK (nivel IN
('Baixo','Médio','Alto','Crítico')),
    id_evento INT NOT NULL REFERENCES
evento(id_evento)
);
```

Criando e Executando o schema.sql

Fatec
Jacareí
Prof. Francisco de Moura

Pelo pgAdmin

- Abra o **pgAdmin**.
- Clique em **Tools** → **Query Tool**.
- Escreva seus comandos SQL (exemplo):

CREATE DATABASE clima_alerta;

- Clique em **Save (Disquete)** e salve o arquivo com o nome: *schema.sql*
- na pasta do projeto *(/scripts/schema.sql*).

- Assim você já tem o script salvo diretamente pelo pgAdmin.

Criando e Executando o schema.sql

Fatec
Jacareí
Prof. Francisco de Moura

Pelo terminal (psql)

- Escreva seus comandos SQL em um editor de texto (ex.: VS Code).
- Salve o arquivo com o nome: *schema.sql*.

- (**Opcional**) Você também pode digitar comandos dentro do psql e usar: \o *schema.sql* para **exportar a saída** para um arquivo.
- Mas essa abordagem é mais usada para exportar resultados, não tanto para escrever schemas.

- O mais comum é escrever o script fora (VS Code) ou direto no pgAdmin.

Executar o schema.sql

Fatec
Jacareí
Prof. Francisco de Moura

- Abra o **pgAdmin**.
- Clique com o botão direito no banco desejado (*clima_alerta*).
- Escolha **Query Tool**.
- Clique em **Open File (ícone de pasta)**.
- Selecione o arquivo *schema.sql*.
- Clique em **Executar (botão play).**

- O pgAdmin vai rodar todos os comandos do script de uma vez só.

Diferenças entre terminal e pgAdmin

Fatec
Jacareí
Prof. Francisco de Moura

- **Terminal (psql)** → mais usado em servidores ou automação (deploys, scripts de inicialização).
- **pgAdmin** → mais usado em ambiente de aprendizado e quando queremos visualizar tudo com interface gráfica.

- Na disciplina, vale a pena aprender **os dois**:
- **pgAdmin** para começar (já estamos usando).
- **psql** para mostrar como é feito no "mundo real" (devops, deploy, servidores).

- Desafio: **Para casa** rodar os exercícios da aula pelo terminal (*psql -f scripts/schema.sql*).

Por que o nome schema.sql?

Fatec
Jacareí
Prof. Francisco de Moura

- A palavra **schema** em Banco de Dados significa **estrutura** (as tabelas, colunas, restrições, relacionamentos etc.).
- Por convenção, muitas equipes chamam o arquivo que contém a **estrutura inicial do banco** de schema.sql.
- **Mas não é obrigatório!** Você poderia chamar de:
  - *clima_alerta.sql*
  - *estrutura.sql*

- Todos funcionariam da mesma forma.

Diferença importante: Schema (conceito do PostgreSQL)

Fatec
Jacareí
Prof. Francisco de Moura

- No PostgreSQL, **schema** também tem outro significado:
  - É como um **"espaço de nomes"** dentro de um banco.
  - Exemplo: no banco *clima_alerta*, existe o *schema* padrão chamado *public*.
- Você poderia criar *schemas* diferentes para organizar tabelas, tipo:
  - *CREATE SCHEMA seguranca;*
  - *CREATE SCHEMA monitoramento;*
- E depois criar tabelas dentro deles:
  - *CREATE TABLE seguranca.usuario (...);*
  - *CREATE TABLE monitoramento.evento (...);*

Ou seja:
•schema.sql → nome de arquivo **convencional** que guarda a estrutura do banco.
•**schema (no PostgreSQL)** → parte do banco que organiza objetos (como tabelas, views, funções).

Diferença importante: Schema (conceito do PostgreSQL)

Fatec
Jacareí
Prof. Francisco de Moura

- Chamamos o arquivo de *schema.sql* por convenção, porque dentro dele fica a **estrutura do banco**.
- Mas vocês poderiam usar outro nome.
- No PostgreSQL, a palavra *schema* também tem outro sentido, que é um **espaço dentro do banco para organizar tabelas**.

- Mais à frente, quando trabalharmos com projetos grandes, vamos ver esse conceito também.

Objetivos da aula

Fatec
Jacareí
Prof. Francisco de Moura

- **Objetivos Gerais:**
  - Conhecer os **comandos essenciais** para consultar informações no banco de dados.
  - Comando **INSERT INTO** (inserindo dados).
  - Comando **SELECT** (consultando dados).
  - Saber como filtrar dados com **WHERE** e ordená-los com **ORDER BY**.
  - Relacionar a consulta de dados com o **desafio da ABP**, para uso prático no projeto.
- **Objetivos Específicos:**
  - Recuperar informações usando **SELECT**.
  - Aplicar filtros para buscar apenas os dados necessários (**WHERE**).
  - Ordenar resultados de forma crescente ou decrescente (**ORDER BY**).
  - Criar consultas que ajudem a resolver **problemas reais** no banco de dados.
  - Usando **Biblioteca** como exemplo genérico.

INSERT INTO — Inserindo Dados

Fatec
Jacareí
Prof. Francisco de Moura

- **Sintaxe básica:**

  INSERT INTO tabela (coluna1, coluna2, ...)

  VALUES (valor1, valor2, ...);

- **Exemplo ABP**

  INSERT INTO tipo_evento (nome, descricao)

  VALUES ('Queimada', 'Incêndio florestal em área de vegetação');

SELECT — Consultando Dados

Fatec Jacareí
Prof. Francisco de Moura

- **Definição:**
  - O Select é o comando usado para **buscar informações** dentro de um banco de dados relacional.
  - Ele permite selecionar **todas as colunas ou apenas algumas** de uma tabela.

- **Sintaxe básica:**
  - *SELECT coluna1, coluna2 FROM nome_da_tabela;*

- **Exemplo (bd_escola):**
  - Buscar todos os alunos da tabela "alunos":
  - SELECT * FROM alunos;

O símbolo * significa **"todas as colunas"**.

Exemplo ABP (clima_alerta):
SELECT nome, descricao FROM tipo_evento;

Filtrando Dados com WHERE

Fatec
Jacareí
Prof. Francisco de Moura

- **Definição:**

  O WHERE permite definir **condições** para filtrar apenas os registros que interessam.

- **Sintaxe básica:**

  SELECT coluna1, coluna2 FROM nome_da_tabela

  WHERE condição;

- **Exemplo (bd_escola):**

  Buscar alunos com idade maior que 20 anos:

  SELECT nome, idade FROM alunos

  WHERE idade > 20;

Agora só veremos os alunos com **mais de 20 anos**!

Exemplo ABP (clima_alerta):
SELECT titulo, status FROM evento WHERE status = 'Ativo';

Filtrando Dados com WHERE

- Outros operadores que podem ser usados no WHERE:

Operador	Significado	Exemplo
`=`	Igual	`idade = 18`
`!=` ou `<>`	Diferente	`curso <> 'ADS'`
`>`	Maior que	`salario > 3000`
`<`	Menor que	`idade < 25`
`LIKE`	Parecido com	`nome LIKE 'J%'` (Nomes que começam com "J")
`BETWEEN`	Intervalo	`data_nasc BETWEEN '2000-01-01' AND '2005-12-31'`

Dica prática

- Inserir poucos registros por vez para **testar**.
- Conferir se a inserção deu certo com *SELECT * FROM tabela*;
- Salvar os inserts em um arquivo separado:
  /scripts/dados_iniciais.sql

Ordenando Resultados com ORDER BY

- **Definição:**

  O ORDER BY é usado para organizar os resultados de uma consulta.

  Pode ordenar **em ordem crescente (ASC)** ou **decrescente (DESC)**.

- **Sintaxe básica:**

  SELECT coluna1, coluna2 FROM nome_da_tabela

  ORDER BY coluna1 ASC;

- **Exemplo:**

  Listar alunos em **ordem alfabética**:

  SELECT nome, idade FROM alunos

  ORDER BY nome ASC;

O padrão é ASC (crescente), mas podemos definir DESC (decrescente).

Combinação de WHERE + ORDER BY

- **Podemos combinar os comandos para filtrar e ordenar ao mesmo tempo!**

- **Exemplo:**
  - Buscar todos os funcionários que ganham mais de R$ 3.000 e ordenar do maior para o menor salário:

```
SELECT nome, cargo, salario FROM funcionarios
WHERE salario > 3000
ORDER BY salario DESC;
```

Agora os **funcionários com maior salário aparecem primeiro!**

Exercícios Individuais

- Continuação dos exercícios: agora vamos popular o banco com dados iniciais.
- Inserir pelo menos **3 registros** em cada tabela principal do projeto *clima_alerta* (ex.: *usuário*, *tipo_evento*, *localizacao*, *evento*).
- Criar **consultas simples** com SELECT em pelo menos 2 tabelas diferentes.
- Criar **consultas filtradas** com WHERE em pelo menos 2 tabelas diferentes.
- Salvar os comandos em: */scripts/dados_iniciais.sql*
- e dar **commit no GitHub individual**.

O que veremos na próxima aula

- Inserções em **múltiplas tabelas** (usando chaves estrangeiras).
- Consultas mais elaboradas com **ORDER BY** e **LIMIT**.
- Continuação do preenchimento de dados do projeto *clima_alerta*.

Referências Bibliográfica da Aula

Fatec
Jacareí
Prof. Francisco de Moura

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

Fatec
Jacareí
Prof. Francisco de Moura

- 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

Fatec
Jacareí
Prof. Francisco de Moura

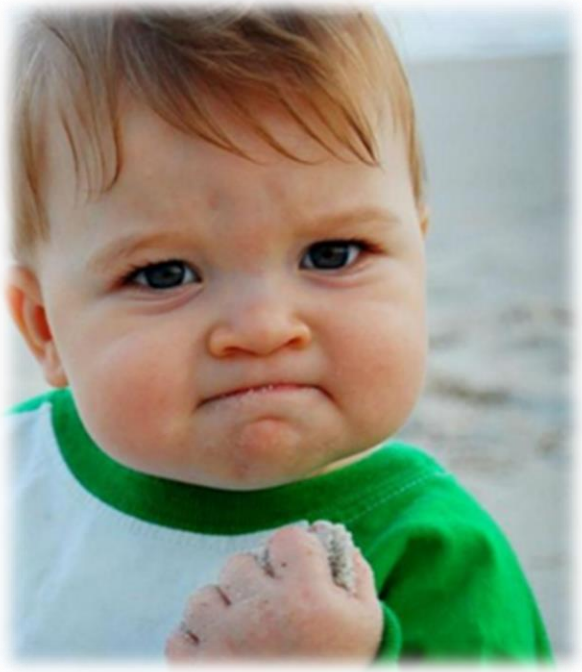
- BEAULIEU, A. **Aprendendo SQL**. São Paulo: Novatec, 2010.
- GILLENSON, M. L. **Fundamentos de Sistemas de Gerência de Banco de Dados**. Rio de Janeiro: LTC, 2006.
- MACHADO, F. N. R. **Banco de Dados:** Projeto e Implementação. São Paulo: Érica, 2005.
- OTEY, M; OTEY, D. **Microsoft SQL Server 2005:** 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: Cencage Learning, 2011.
- TEOREY, T; LIGHTSTONE, S; NADEAU, T. **Projeto e Modelagem de Bancos de Dados.** São Paulo: Campus, 2006.

Dúvidas?

Fatec
Jacareí
Prof. Francisco de Moura

Considerações Finais

Fatec
Jacareí
Prof. Francisco de Moura



Professora:

Lucineide Pimenta

Bom descanso à todos!