PostgreSQL e pgAdmin

Aula 10 — Do papel para o banco de verdade

Aula 10 · 21/10/2026 · Indicadores 1, 3

UniSenac Centro Universitário RS
Senac
Fecomércio
IFEP
Sesc
Sindicatos Empresariais

O mapa de hoje

- **Bloco 1** — instalar o PostgreSQL e o pgAdmin, passo a passo, juntos
- **Bloco 2** — tour do pgAdmin e o que é DDL
- **Bloco 3** — `CREATE TABLE`: a anatomia do comando
- **Bloco 4** — mão na massa: criar o banco e a **primeira tabela real** do projeto

> Hoje é a aula com mais chance de dar problema técnico. **Levantem a mão cedo** — ficar travado 20 minutos em silêncio é o pior cenário.

O que a gente vai instalar (e por quê são dois)



- **PostgreSQL** — o servidor. Fica ligado sozinho, guarda os dados, faz valer as regras
- **pgAdmin** — o cliente. A janela para conversar com o servidor

> Fechar o pgAdmin **não** desliga o banco. São dois programas separados — lembram da Aula 02?

Bloco 1

Instalação guiada

Passo a passo — baixar

1. Abram o navegador em **postgresql.org/download/windows**
2. Cliquem em **"Download the installer"**
3. Escolham a versão **16** para **Windows x86-64**
4. Esperem o download (cerca de 350 MB — vai demorar)

> No laboratório o instalador **já está** na pasta compartilhada da turma — não baixem, copiem de lá. Só quem estiver no notebook próprio precisa baixar.

A decisão mais importante da instalação: a senha

O instalador vai pedir uma **senha para o usuário** `postgres`. Essa senha **não tem como ser recuperada**. Se você esquecer, a saída é desinstalar e instalar tudo de novo.

1. Use uma senha **simples** — é um banco de estudo na sua máquina
2. **Anote** agora, no celular ou no caderno, antes de digitar
3. No laboratório, a senha é a que está no quadro — para todo mundo igual

Passo a passo — instalar

1. Executem o instalador e cliquem **Next** até a tela de componentes
2. Deixem marcados: **PostgreSQL Server**, **pgAdmin 4** e **Command Line Tools**
3. Podem **desmarcar** o Stack Builder — não vamos usar
4. Aceitem a pasta padrão de dados
5. Digitem a **senha** (a do quadro) — duas vezes
6. Porta: deixem **5432**. Locale: deixem o padrão
7. **Next** até o fim e esperem (uns 5 minutos)

⚠ Se der problema

"Porta 5432 já em uso": já existe um PostgreSQL instalado. Me chame — não instale outro por cima.
Antivírus bloqueou: normal. Autorize, ou me chame.
"Sem permissão de administrador": nas máquinas do laboratório eu tenho a senha de admin.
Instalou e o pgAdmin não abre: espere. A primeira abertura demora até 2 minutos.
Travou de vez: use a máquina do laboratório hoje e a gente resolve o seu notebook no intervalo. **Não perca a aula por causa da instalação.**

Checagem do Bloco 1

Antes do intervalo, confirme na sua tela:

- O **pgAdmin abre** e mostra "Servers" na lateral esquerda?
- Ao clicar na setinha de "Servers", **pede a senha**?
- Digitando a senha, **abre** e aparece "PostgreSQL 16"?

> Se as três respostas forem "sim", você está pronto. Se alguma for "não", **levante a mão agora**, antes do intervalo.

forms.gle/cpabVDxqWwFhQ6vM6

Intervalo

10 minutos

Bloco 2

Tour do pgAdmin e o que é DDL

A hierarquia do pgAdmin

O CAMINHO ATÉ AS SUAS TABELAS

1 · Servers ▸ 2 · PostgreSQL 16 ▸ 3 · Databases

4 · locadora ▸ 5 · Schemas ▸ 6 · public

7 · Tables ▸ 8 · suas tabelas ▸

O caminho é longo e é fácil parar no meio. Decore a sequência: **Servers** → **o servidor** → **Databases** → **o seu banco** → **Schemas** → **public** → **Tables**.

Criar o banco do projeto

1. Botão direito em **Databases** → **Create** → **Database...**
2. Em **Database**, digite o nome: `locadora` (ou o tema do seu grupo)
3. Deixe **Owner** como `postgres`
4. Clique em **Save**
5. O banco aparece na lista da esquerda

> Nome do banco: minúsculas, sem acento e sem espaço — as mesmas regras dos nomes de coluna da Aula 04.

O Query Tool — onde o SQL acontece

1. Clique no **seu banco** para selecioná-lo
2. Menu **Tools** → **Query Tool** (ou o ícone de raio)
3. A janela de cima é onde você **escreve** o SQL
4. O botão ▶ (ou **F5**) executa
5. A janela de baixo mostra o **resultado** ou o **erro**

> **Confira sempre o nome do banco no topo da janela.** Escrever SQL no banco errado é o erro mais comum — e o mais confuso, porque o comando funciona e a tabela "some".

O que é DDL

GLOSSÁRIO

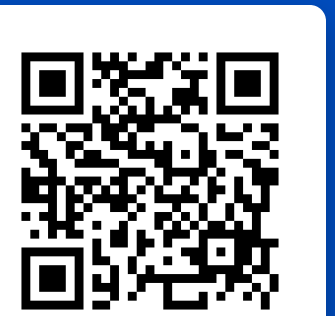
DDL	Data Definition Language	Linguagem de Definição de Dados	criar e alterar a **estrutura**

DDL mexe no **esquema** — a forma, aquilo que quase nunca muda (Aula 02).

Comando	EN → PT
`CREATE TABLE`	criar tabela
`ALTER TABLE`	alterar tabela
`DROP TABLE`	remover tabela

Checagem do Bloco 2

- Qual é o caminho completo, do "Servers" até uma tabela?
- Onde você confere em qual **banco** o Query Tool está?
- `CREATE TABLE` é DDL ou DML? E `INSERT`?



forms.gle/x6EmAVSPHvQVhcXS7

Intervalo

10 minutos

Bloco 3

CREATE TABLE — a anatomia do comando

O comando inteiro, e o que cada parte faz

```
CREATE TABLE plataforma (
    id_plataforma  SERIAL       PRIMARY KEY,
    nome           VARCHAR(50)  NOT NULL,
    fabricante     VARCHAR(50)
);
```

GLOSSÁRIO

`CREATE TABLE`	criar tabela
`SERIAL`	série — o banco gera o número sozinho (1, 2, 3…)
`PRIMARY KEY`	chave primária
`VARCHAR(n)`	variable character — texto de até n caracteres
`NOT NULL`	não nulo — a coluna nunca pode ficar vazia

Do papel para o SQL — a tradução é direta

No esquema (Aula 09)	No SQL
`plataforma (...)`	`CREATE TABLE plataforma (...)`
`id_plataforma*`	`id_plataforma SERIAL PRIMARY KEY`
`nome` (obrigatório)	`nome VARCHAR(50) NOT NULL`
`fabricante` (pode ser nulo)	`fabricante VARCHAR(50)`

A coluna "Obrigatório?" do **dicionário de dados** (Aula 04) vira `NOT NULL`. Foi por isso que a gente preencheu aquela coluna.

SERIAL — o número que o banco gera

```sql
INSERT INTO plataforma (nome, fabricante) VALUES ('PlayStation 5', 'Sony');
-- id_plataforma vira 1, sozinho

INSERT INTO plataforma (nome, fabricante) VALUES ('Nintendo Switch', 'Nintendo');
-- id_plataforma vira 2, sozinho
```

Com `SERIAL`, você **nunca** digita o valor da chave primária. O banco cuida disso — e garante que não repete.

GLOSSÁRIO

`AUTO_INCREMENT` `SERIAL`

ALTER e DROP — mudar e apagar

```sql
ALTER TABLE jogo ADD COLUMN ano_lancamento INTEGER;

ALTER TABLE jogo DROP COLUMN ano_lancamento;

DROP TABLE jogo;
```

`DROP TABLE` **apaga a tabela e todos os dados dentro dela, sem perguntar nada.** Não existe "desfazer". Antes de digitar DROP, leia duas vezes o nome da tabela.

⚠ Erros que o Postgres vai mostrar hoje

Mensagem	O que houve
`syntax error at or near ")"`	faltou ou sobrou vírgula antes do parêntese
`relation "plataforma" already exists`	a tabela já foi criada; use outro nome ou `DROP` antes
`type "varchar" does not exist`	erro de digitação no tipo
`unterminated quoted string`	faltou fechar a aspa simples

Leia a mensagem de erro **inteira**. O PostgreSQL diz a linha e a coluna onde travou — quase sempre o erro está um pouco antes do ponto indicado.

Checagem do Bloco 3

O que está errado neste comando?

```
CREATE TABLE cliente (
    id_cliente SERIAL PRIMARY KEY,
    nome VARCHAR(100) NOT NULL,
    telefone VARCHAR(20),
)
```

forms.gle/zb6GjkPn5SAUBrkS7

Intervalo

10 minutos

MÃO NA MASSA

Criar o banco e a primeira tabela do projeto

1. Confirmem que o banco do grupo existe no pgAdmin
2. Abram o **Query Tool** — confiram o nome do banco no topo
3. Abram o **esquema normalizado** da Aula 09 ao lado
4. Escolham uma tabela **sem chave estrangeira** para ser a primeira
5. Escrevam o `CREATE TABLE`, traduzindo do esquema linha por linha
6. Executem com **F5** e leiam a mensagem
7. Confiram em **Tables** que a tabela apareceu (botão direito → **Refresh**)

MÃO NA MASSA

O combinado desta entrega

GLOSSÁRIO

`grupoNN-aulaNN-artefato.<ext>` `grupo03-aula10-criar-tabelas.sql`

O que eu confiro: O Query Tool está no **banco do grupo**, e não em `postgres` · A chave primária usa `SERIAL PRIMARY KEY` · As colunas obrigatórias do dicionário estão com `NOT NULL` · Os nomes seguem as **4 regras** da Aula 04 · O `VARCHAR(n)` tem um `n` **pensado**, e não chutado

> Terminou? Crie mais uma tabela sem chave estrangeira. Amanhã a gente liga todas.

PARA CASA · OPCIONAL

Antes da próxima aula

Testar em casa: instale o PostgreSQL e o pgAdmin no seu notebook pessoal e confirme que abre. **Anote a senha.**

Tarefa opcional: crie uma tabela de teste qualquer (`favoritos`, `filmes`, o que quiser), fora do projeto, só para praticar o `CREATE TABLE` sem medo de estragar nada.

Para treinar: `locadora-exercicios-10-criar.sql`, na página da aula — oito situações, metade delas para dar erro de propósito.

Para ver acontecer: demonstrações da Aula 10 — desligar o servidor e ver o pgAdmin falhar, e descer a árvore até achar a tabela — e oito perguntas com resposta na hora. No site da UC, na página da aula.

Para a próxima aula

As tabelas existem, mas estão **soltas**. Como fazer o banco garantir que um empréstimo não seja de um cliente inexistente?

UniSenac Centro Universitário RS
Senac
Fecomércio
IFEP
Sesc
Sindicatos Empresariais