

Criando Bancos e Tabelas no PostgreSQL

14.5

Aula 5

Introdução a linguagem SQL

3

DDL, DML, DCL

SQL

Sub-linguagens do SQL



Vamos estudar algumas “sub-linguagens” do SQL:

- **Linguagem de Definição de Dados – DDL**
- Linguagem de Manipulação de Dados – DML
- Linguagem de Controle de Dados - DCL

Linguagem de Definição de Dados - DDL

Objetos
fundamentais
da
linguagem:

DATABASE
USER
ROLE
SCHEMA
TABLESPACE
TABLE
INDEX
DOMAIN
FUNCTION
SEQUENCE
TRIGGER
VIEW

Todos os objetos podem ser criados (CREATE), corrigidos (ALTER) e removidos (DROP), cada um com sintaxe própria.

DDL - Comando CREATE TABLE

Criar uma Tabela no Esquema da Aplicação

Sintaxe geral de um Comando CREATE TABLE

```
CREATE TABLE <nome da tabela> (  
    <definição de Coluna>,...  
    <Restrições de Integridade>,...  
);
```

onde <definição de Coluna> pode ser:

```
<nome atr> <tipo de dado> [NULL |  
    NOT NULL]  
[DEFAULT <value>] — DEFAULT NULL
```

Exemplo

Curso de Banco de Dados (USP)

Aluno={NUSP, Nome, Cidade, Idade}

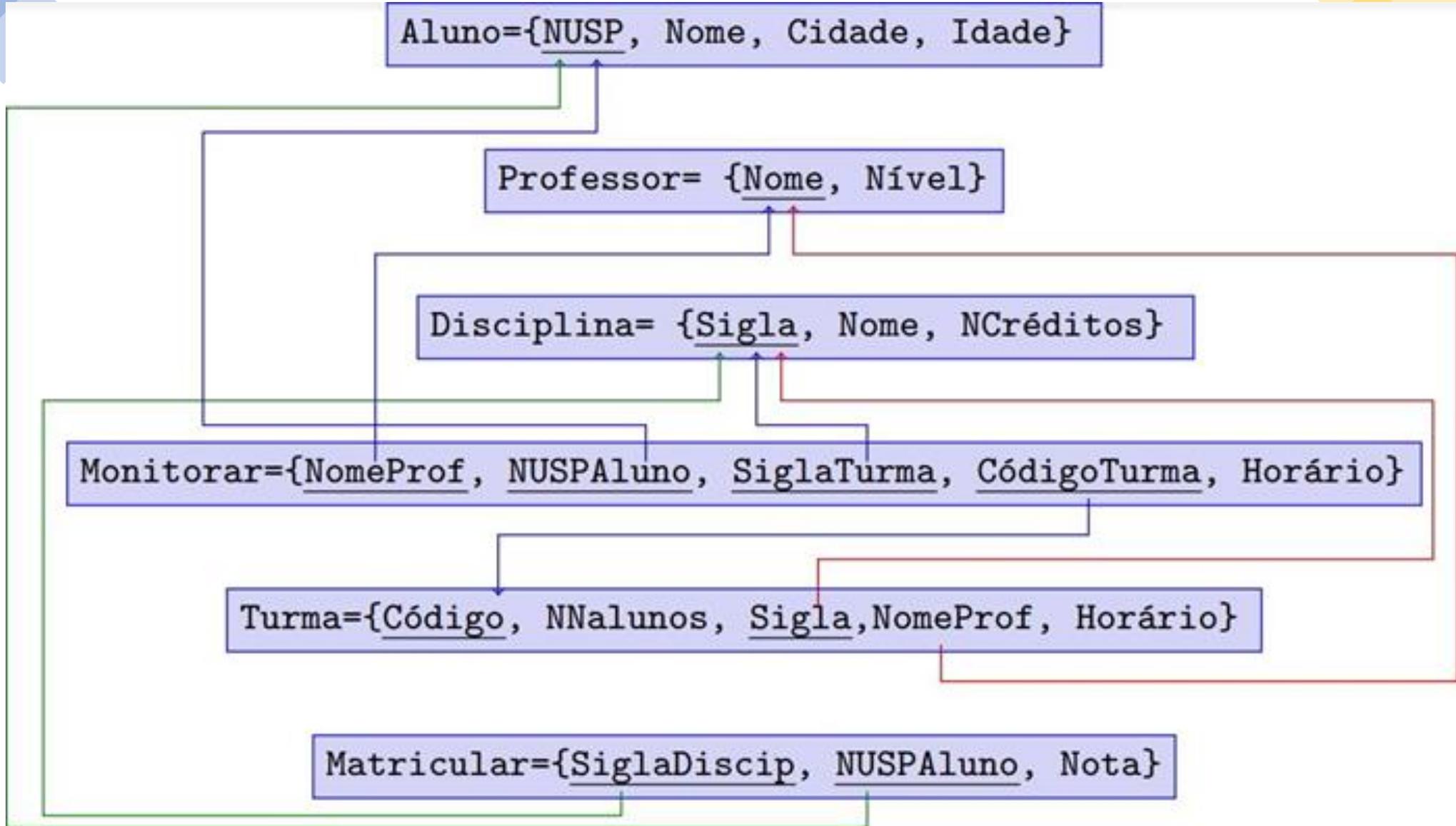
Professor= {Nome, Nível}

Disciplina= {Sigla, Nome, NCréditos}

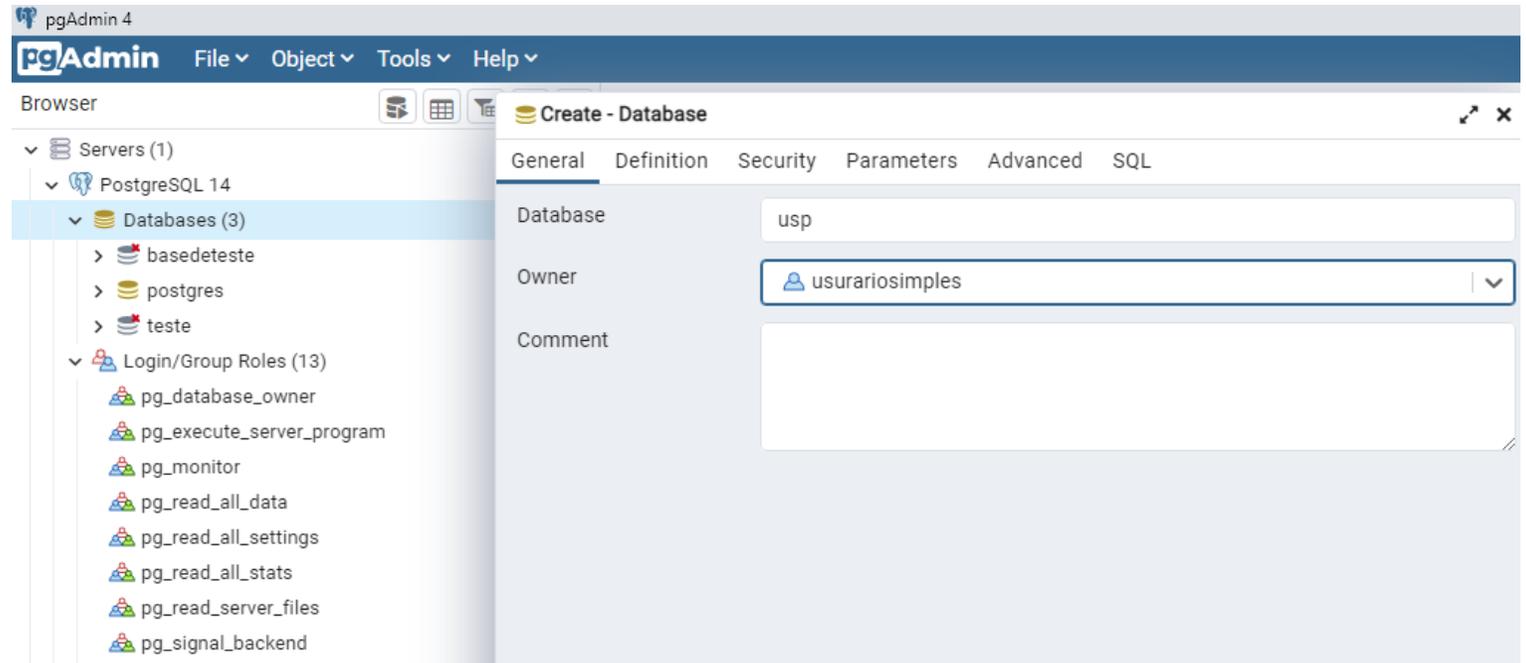
Monitorar={NomeProf, NUSPALuno, SiglaTurma, CódigoTurma, Horário}

Turma={Código, NNalunos, Sigla, NomeProf, Horário}

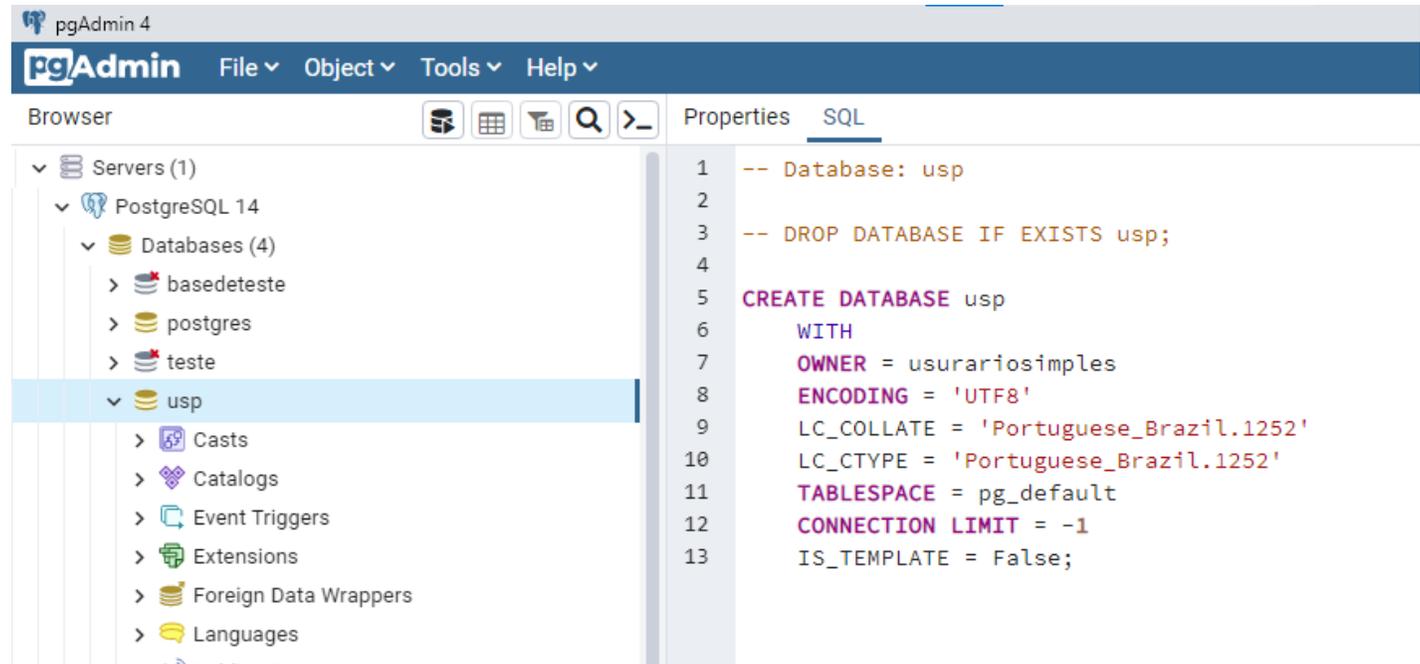
Matricular={SiglaDiscip, NUSPALuno, Nota}



Criar a base de dados usp



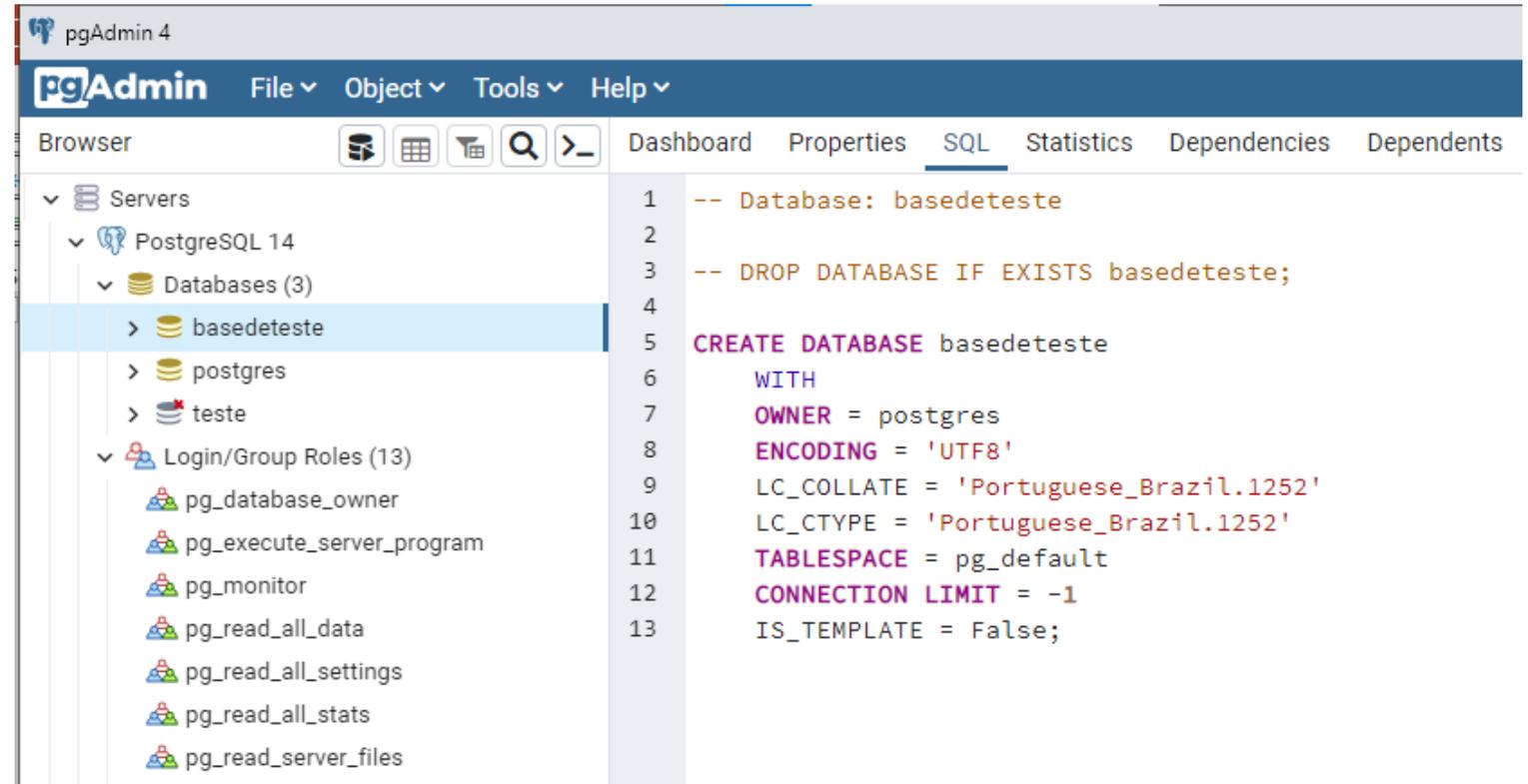
Criar a base de dados usp



The screenshot shows the pgAdmin 4 interface. The left pane displays the server tree with 'PostgreSQL 14' expanded to show 'Databases (4)'. The 'usp' database is selected and highlighted. The right pane shows the SQL editor with the following SQL code:

```
1  -- Database: usp
2
3  -- DROP DATABASE IF EXISTS usp;
4
5  CREATE DATABASE usp
6  WITH
7  OWNER = usuariosimples
8  ENCODING = 'UTF8'
9  LC_COLLATE = 'Portuguese_Brazil.1252'
10 LC_CTYPE = 'Portuguese_Brazil.1252'
11 TABLESPACE = pg_default
12 CONNECTION LIMIT = -1
13 IS_TEMPLATE = False;
```

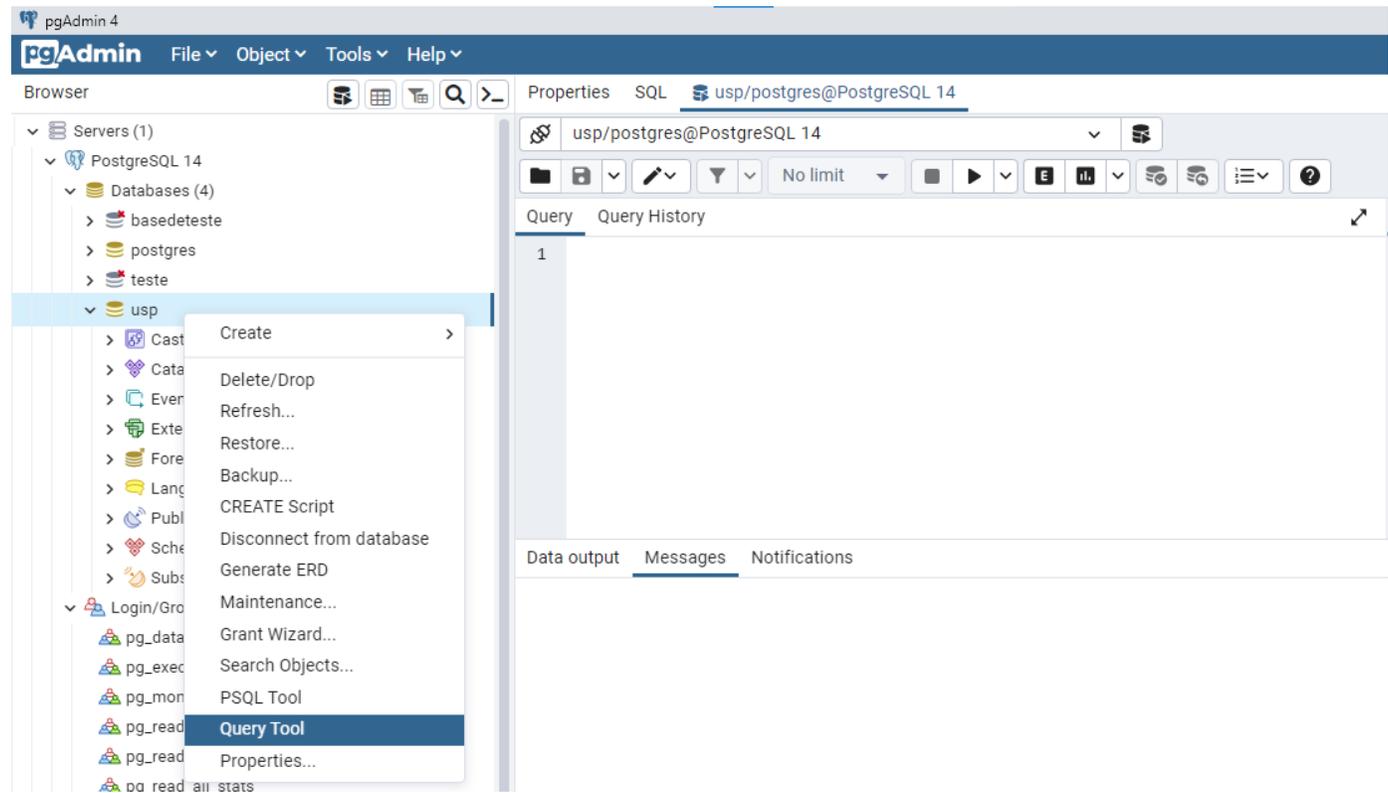
Situação
após uma
conexão:



The screenshot shows the pgAdmin 4 interface. The left sidebar displays a tree view of the database structure. Under 'Servers', 'PostgreSQL 14' is expanded to show 'Databases (3)', with 'basedeteste' selected. Below it are 'postgres' and 'teste'. Under 'Login/Group Roles (13)', several roles are listed, including 'pg_database_owner', 'pg_execute_server_program', 'pg_monitor', 'pg_read_all_data', 'pg_read_all_settings', 'pg_read_all_stats', and 'pg_read_server_files'. The main window shows the 'SQL' tab with the following SQL query:

```
1  -- Database: basedeteste
2
3  -- DROP DATABASE IF EXISTS basedeteste;
4
5  CREATE DATABASE basedeteste
6  WITH
7  OWNER = postgres
8  ENCODING = 'UTF8'
9  LC_COLLATE = 'Portuguese_Brazil.1252'
10 LC_CTYPE = 'Portuguese_Brazil.1252'
11 TABLESPACE = pg_default
12 CONNECTION LIMIT = -1
13 IS_TEMPLATE = False;
```

Situação
após uma
conexão:



Exemplo:
Projeto
Lógico: Uma
tabela é
descrita
indicando
seus
atributos
(com as
respectivas
restrições de
integridade):

- Aluno={NUSP, Nome, Cidade, Idade}

```
CREATE TABLE Aluno (  
    NUSP decimal(7) NOT NULL,  
    Nome varchar(60) NOT NULL,  
    Cidade char(25),  
    Idade decimal(3) NOT NULL  
);
```

Onde?

- ▼ usp
 - > Casts
 - > Catalogs
 - > Event Triggers
 - > Extensions
 - > Foreign Data Wrappers
 - > Languages
 - > Publications
 - ▼ Schemas (1)
 - ▼ public
 - > Aggregates
 - > Collations
 - > Domains
 - > FTS Configurations
 - > FTS Dictionaries
 - > FTS Parsers
 - > FTS Templates
 - > Foreign Tables
 - > Functions
 - > Materialized Views
 - > Operators
 - > Procedures
 - > Sequences
 - ▼ Tables (1)
 - ▼ aluno

- ▼ Tables (1)
 - ▼ aluno
 - ▼ Columns (4)
 - nusp
 - nome
 - cidade
 - idade
 - > Constraints

Visualizar:

Properties SQL usp_alunos.sql*

usp/postgres@PostgreSQL 14

No limit

Query Query History

```
1 SELECT * FROM Aluno;  
2
```

Data output Messages Notifications

nusp numeric (7)	nome character varying (60)	cidade character (25)	idade numeric (3)
---------------------	--------------------------------	--------------------------	----------------------

Exemplo:
Criar uma
tabela com
Atributos
DEFAULT

- Professor= {Nome, Nível}

```
CREATE TABLE Professor (  
    Nome varchar(60) NOT NULL,  
    Nivel char(4) NOT NULL DEFAULT 'MS-S'  
);
```

Onde?

- Tables (2)
 - aluno
 - Columns (4)
 - Constraints
 - Indexes
 - RLS Policies
 - Rules
 - Triggers
 - professor
 - Columns (2)
 - nome
 - nivel
 - Constraints
 - Indexes

...DDL

SQL

DDL - Comando CREATE TABLE

tipo de
dato
pode
ser:

**SMALLINT | INTEGER | BIGINTEGER | FLOAT |
DOUBLE PRECISION
| {DECIMAL | NUMERIC}[(precision [, scale])]
| DATE | TIME | TIMESTAMP
| {CHAR | CHARACTER | CHARACTER VARYING
| VARCHAR}[(int)]
| CLOB | BLOB**

DDL - Comando CREATE TABLE

Restrições de Integridade

PRIMARY KEY [<nome da chave>] (<Atributo>,...)
| UNIQUE [<nome da chave>] (<Atributo>,...)
| FOREIGN KEY [<nome da chave>]
(<Atributo>,...>)
REFERENCES <tabela> (<Atributos
chaves>, ...)[<ações>]
| CHECK [<nome da condição>] (<condição>)

DDL - Comando CREATE TABLE

onde
<ações>
pode
ser:

ON UPDATE
I DELETE CASCADE
I RESTRICT
I SET NULL
I SET DEFAULT
I NO ACTION

...Exemplo

Curso de Banco de Dados (USP)

Exemplo:
Criar uma
tabela com
as
Restrições
de
Integridade,
como
restrição de
atributo

- Aluno={NUSP, Nome, Cidade, Idade}

```
CREATE TABLE Aluno (  
    NUSP decimal(7) NOT NULL PRIMARY KEY,  
    Nome varchar(60) NOT NULL,  
    Cidade char(25),  
    Idade decimal(3) NOT NULL  
);
```

OU

...

- Aluno={NUSP, Nome, Cidade, Idade}

```
CREATE TABLE Aluno (  
    NUSP decimal(7) NOT NULL,  
    Nome varchar(60) NOT NULL,  
    Cidade char(25),  
    Idade decimal(3) NOT NULL,  
    PRIMARY KEY (NUSP)
```

```
);
```

Refazer aluno

The image shows a screenshot of a database query editor interface. At the top, there is a toolbar with icons for file operations (folder, save), editing (pencil), filtering (funnel), and execution (play, stop, refresh). Below the toolbar, there are two tabs: "Query" and "Query History". The "Query" tab is active, showing a single line of SQL code: "1 DROP TABLE aluno;". Below the query editor, there are three tabs: "Data output", "Messages", and "Notifications". The "Messages" tab is active, displaying the output of the query: "DROP TABLE" followed by "Query returned successfully in 238 msec."

Query Query History

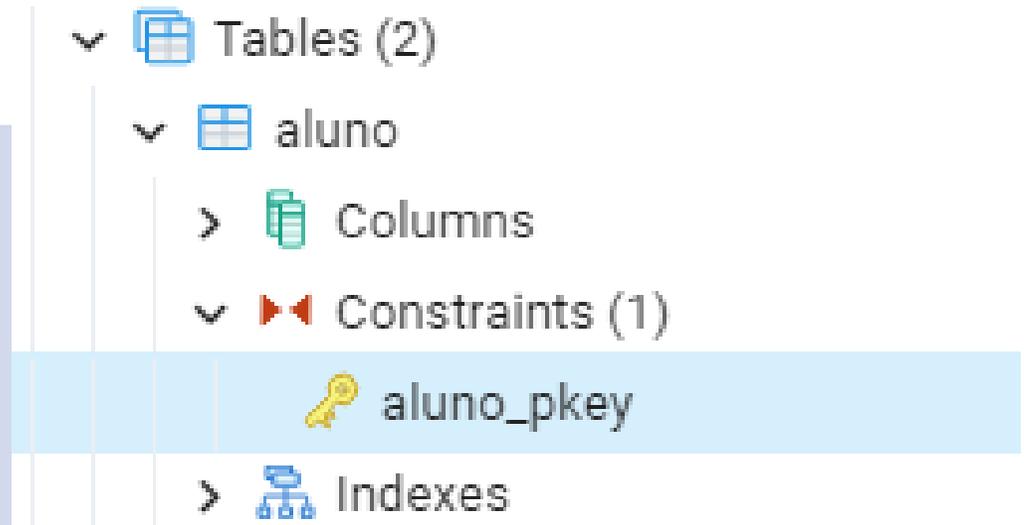
```
1 DROP TABLE aluno;
```

Data output Messages Notifications

DROP TABLE

Query returned successfully in 238 msec.

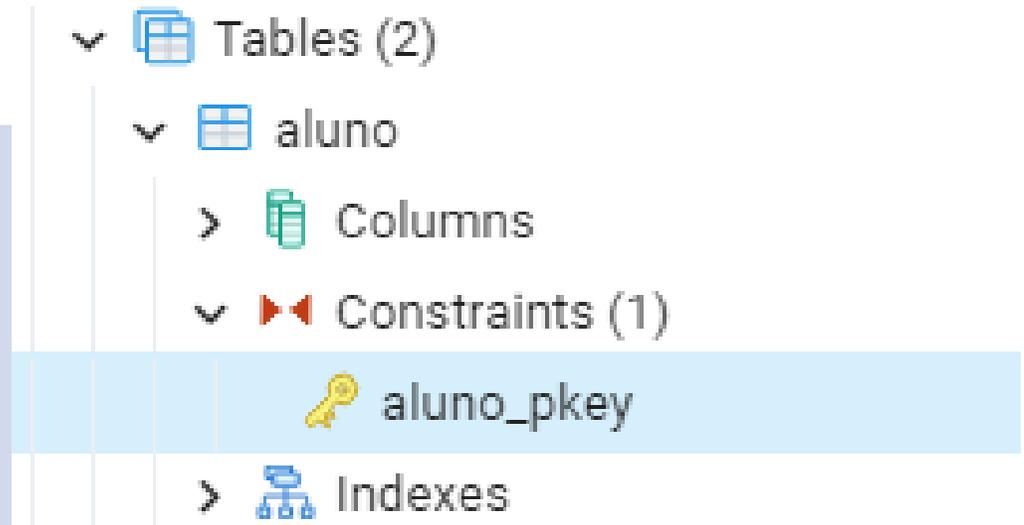
Exemplo:
Criar uma
tabela com
as
Restrições
de
Integridade,
como
restrição de
atributo



- Aluno={NUSP, Nome, Cidade, Idade}

```
CREATE TABLE Aluno (  
    NUSP decimal(7) NOT NULL PRIMARY KEY,  
    Nome varchar(60) NOT NULL,  
    Cidade char(25),  
    Idade decimal(3) NOT NULL  
);
```

Exemplo:
Criar uma
tabela com
as
Restrições
de
Integridade,
como
restrição de
atributo



- Aluno={NUSP, Nome, Cidade, Idade}

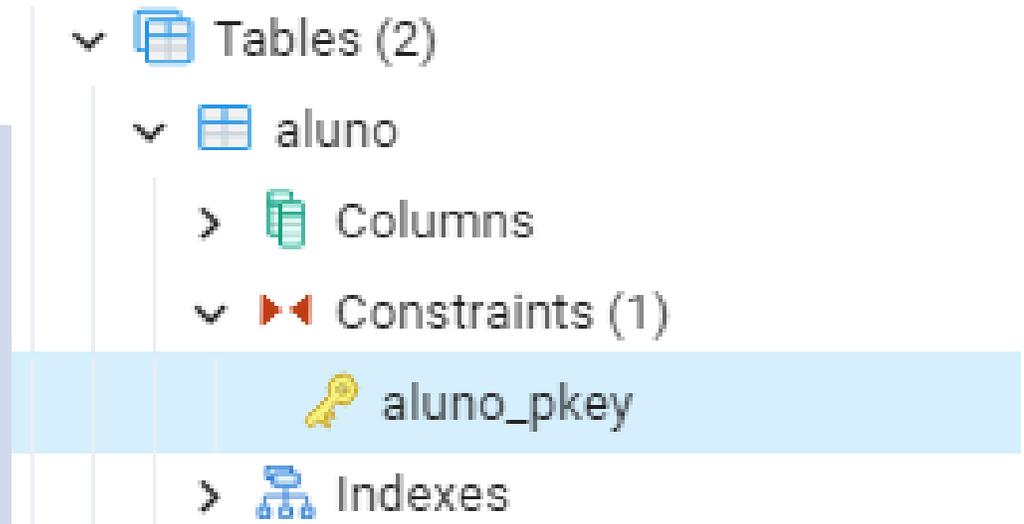
```
CREATE TABLE Aluno (  
    NUSP decimal(7) NOT NULL PRIMARY KEY,  
    Nome varchar(60) NOT NULL,  
    Cidade char(25),  
    Idade decimal(3) NOT NULL  
);
```

Disciplina= {Sigla, Nome, NCréditos}

Turma={Código, NNalunos, Sigla, Número, NomeProf, Horário}

Novo esquema...

Exemplo:
Criar uma
tabela com
as
Restrições
de
Integridade,
como
restrição de
atributo



- Disciplina= {Sigla, Nome, NCreditos}

```
CREATE TABLE disciplina (  
    Sigla DECIMAL(4) NOT NULL PRIMARY KEY,  
    Nome VARCHAR(60) NOT NULL,  
    ncreditos DECIMAL (2)  
);
```

Resultado

The screenshot displays a PostgreSQL database management interface. On the left, a tree view shows the database structure, with the 'disciplina' table highlighted under the 'Tables (3)' folder. The main window shows a SQL query editor with the following code:

```
1 CREATE TABLE disciplina (  
2     Sigla DECIMAL(4) NOT NULL PRIMARY KEY,  
3     Nome VARCHAR(60) NOT NULL,  
4     NCREDITOS DECIMAL (2)  
5 );  
6  
7
```

Below the query editor, the 'Messages' tab is active, showing the execution result:

```
CREATE TABLE  
  
Query returned successfully in 139 msec.
```

Popular a base com Dados

Exemplo CVS

Fontes

Formula 1

- <http://ergast.com/mrd/db/> Last updated: 12/06/2018

Alternativas

- <https://www.kaggle.com/cjgdev/formula-1-race-data-19502017>
- <https://www.kaggle.com/datasets/rohanrao/formula-1-world-championship-1950-2020>

Aeroportos:

- <https://ourairports.com/data/>

GeoCities:

- <https://www.geonames.org/export/>

Not secure | ergast.com/mvd/db/

API do Desenvolvedor Ergast

Banco de Imagens

As imagens completas do banco de dados Ergast são publicadas logo após cada corrida sob a licença Attribution-NonCommercial-ShareAlike 3.0 Unported .



A estrutura do banco de dados é mostrada em um Diagrama de Relacionamento de Entidade e explicada no Guia do Usuário do Banco de Dados . As tabelas foram criadas usando: f1db_tables.sql

As imagens do banco de dados foram atualizadas pela última vez em: 14/11/2022

Imagens do banco de dados MySQL

Estes são despejos de dados do MySQL (5.7). A versão ANSI pode ser mais fácil de importar para outros formatos de banco de dados. A codificação de caracteres é "UTF8" em ambas as versões.

[f1db.sql.gz](#) Despejo do banco de dados MySQL 5.7

[f1db_ansi.sql.gz](#) Despejo de banco de dados MySQL "compatível com ANSI"

Tabelas de banco de dados CSV

O banco de dados também pode ser baixado como um conjunto de arquivos CSV que podem ser importados para planilhas e outros tipos de software:

[f1db_csv.zip](#)

Cada arquivo CSV contém uma única tabela de banco de dados. A codificação de texto é UTF-8 e a primeira linha de cada arquivo contém os cabeçalhos das colunas. As tabelas são descritas no Guia do Usuário .

Isenção de responsabilidade

O autor se isenta de qualquer responsabilidade por qualquer perda ou dano decorrente do uso dessas imagens de banco de dados e não pode garantir a precisão dos dados.

Atribuição

A atribuição é opcional, mas apreciada ao abordar um público técnico. Uma referência a "Ergast" com um link para este site é suficiente.

Índice

- Documentação da API
- Lista de Temporadas
- Programa de Corrida
- Resultados de corrida
- Resultados de qualificação
- Resultados de qualificação de sprint
- Classificação
- Informações do motorista
- Informações do Construtor
- Informações do Circuito
- Status de acionamento
- Tempos de volta
- F1 Stops
- Parâmetros de desenvolvimento
- banco de dados de consulta
- Banco de Imagens
- Galeria de aplicativos
- termos e Condições
- Perguntas Frequentes
- Comércio
- Últimas notícias
- Relatório de erros

links

- Contato-nos
- Web empresarial

meta

- Conectar-se
- Entrada RSS
- RSS de comentários
- WordPress.org

Procurar por:



drivers	
driverId	int(11)
driverRef	varchar(255)
number	int(11)
code	varchar(3)

seasons	
year	int(11)
url	varchar(255)

qualifying	
qualifyId	int(11)
raceId	int(11)
driverId	int(11)
constructorId	int(11)
number	int(11)
position	int(11)
q1	varchar(255)
q2	varchar(255)
q3	varchar(255)

results	
resultId	int(11)
raceId	int(11)
driverId	int(11)
constructorId	int(11)
number	int(11)
grid	int(11)
position	int(11)
positionText	varchar(255)
positionOrder	int(11)
points	float
laps	int(11)
time	varchar(255)
milliseconds	int(11)
fastestLap	int(11)
rank	int(11)
fastestLapTime	varchar(255)
fastestLapSpeed	varchar(255)
statusId	int(11)

driverStandings	
driverStandingsId	int(11)
raceId	int(11)
driverId	int(11)
points	float
position	int(11)
positionText	varchar(255)
wins	int(11)

constructorStandings	
constructorStandingsId	int(11)
raceId	int(11)
constructorId	int(11)
points	float
position	int(11)
positionText	varchar(255)
wins	int(11)

sprintResults	
sprintResultId	int(11)
raceId	int(11)
driverId	int(11)
constructorId	int(11)
number	int(11)
grid	int(11)
position	int(11)
positionText	varchar(255)
positionOrder	int(11)
points	float
laps	int(11)
time	varchar(255)
milliseconds	int(11)
fastestLap	int(11)
fastestLapTime	varchar(255)
statusId	int(11)

pitStops	
raceId	int(11)
driverId	int(11)
stop	int(11)
lap	int(11)
time	time
duration	varchar(255)
milliseconds	int(11)

constructorResults	
constructorResultsId	int(11)
raceId	int(11)
constructorId	int(11)
points	float
status	varchar(255)

status	
statusId	int(11)
status	varchar(255)

lapTimes	
raceId	int(11)
driverId	int(11)
lap	int(11)
position	int(11)
time	varchar(255)
milliseconds	int(11)

circuits	
circuitId	int(11)
circuitRef	varchar(255)
name	varchar(255)
location	varchar(255)
country	varchar(255)
lat	float
lng	float
alt	int(11)
url	varchar(255)



List of Tables
circuits
constructorResults
constructorStandings
constructors
driverStandings
drivers
lapTimes
pitStops
qualifying
races
results
seasons
status

General Notes
Dates, times and durations are in ISO 8601 format
Dates and times are UTC
Strings use UTF-8 encoding
Primary keys are for internal use only
Fields ending with "Ref" are unique identifiers for external use
A grid position of '0' is used for starting from the pitlane
Labels used in the positionText fields:
"D" - disqualified
"E" - excluded
"F" - failed to qualify
"N" - not classified
"R" - retired
"W" - withdrew

circuits table

Field	Type	Null	Key	Default	Extra	Description
circuitId	int(11)	NO	PRI	NULL	auto_increment	Primary key
circuitRef	varchar(255)	NO				Unique circuit identifier
name	varchar(255)	NO				Circuit name
location	varchar(255)	YES		NULL		Location name
country	varchar(255)	YES		NULL		Country name
lat	float	YES		NULL		Latitude
lng	float	YES		NULL		Longitude

circuits.csv

Arquivo Editar Formatar Exibir Ajuda

```
circuitId,circuitRef,name,location,country,lat,lng,alt,url
```

```
1,"albert_park","Albert Park Grand Prix Circuit","Melbourne","Australia",-37.8497,144.
2,"sepang","Sepang International Circuit","Kuala Lumpur","Malaysia",2.76083,101.738,18
3,"bahrain","Bahrain International Circuit","Sakhir","Bahrain",26.0325,50.5106,7,"http
4,"catalunya","Circuit de Barcelona-Catalunya","Montmeló","Spain",41.57,2.26111,109,"l
5,"istanbul","Istanbul Park","Istanbul","Turkey",40.9517,29.405,130,"http://en.wikipe
6,"monaco","Circuit de Monaco","Monte-Carlo","Monaco",43.7347,7.42056,7,"http://en.wil
7,"villeneuve","Circuit Gilles Villeneuve","Montreal","Canada",45.5,-73.5228,13,"http
8,"magny_cours","Circuit de Nevers Magny-Cours","Magny Cours","France",46.8642,3.1636:
9,"silverstone","Silverstone Circuit","Silverstone","UK",52.0786,-1.01694,153,"http://
10,"hockenheimring","Hockenheimring","Hockenheim","Germany",49.3278,8.56583,103,"http
11,"hungaroring","Hungaroring","Budapest","Hungary",47.5789,19.2486,264,"http://en.wil
12,"valencia","Valencia Street Circuit","Valencia","Spain",39.4589,-0.331667,4,"http:,
13,"spa","Circuit de Spa-Francorchamps","Spa","Belgium",50.4372,5.97139,401,"http://er
14,"monza","Autodromo Nazionale di Monza","Monza","Italy",45.6156,9.28111,162,"http://
15,"marina_bay","Marina Bay Street Circuit","Marina Bay","Singapore",1.2914,103.864,18
16,"fuji","Fuji Speedway","Oyama","Japan",35.3717,138.927,583,"http://en.wikipedia.org
17,"shanghai","Shanghai International Circuit","Shanghai","China",31.3389,121.22,5,"ht
18,"interlagos","Autódromo José Carlos Pace","São Paulo","Brazil",-23.7036,-46.6997,78
19,"indianapolis","Indianapolis Motor Speedway","Indianapolis","USA",39.795,-86.2347,:
20,"nurburgring","Nürburgring","Nürburg","Germany",50.3356,6.9475,578,"http://en.wikip
21,"imola","Autodromo Enzo e Dino Ferrari","Imola","Italy",44.3439,11.7167,37,"http://
22,"suzuka","Suzuka Circuit","Suzuka","Japan",34.8431,136.541,45,"http://en.wikipedia
23 "osterreichring" "A1-Ring" "Spielberg" "Austria" 47 2197 14 7647 678 "http://en wil
```

Browser



- > Extensions (1)
- > Foreign Data Wrappers
- > Languages
- > Publications
- > Schemas (1)

public

- Create >
- Delete/Drop
- Refresh...
- Restore...
- Backup...
- Drop Cascade
- CREATE Script
- Grant Wizard...

Properties SQL usp_alunos.sql*

usp/postgres@PostgreSQL 14

File Save Filter No limit Run

Query Query History

```
1 CREATE DATABASE formula1;
```

Circuitos

Script de
criação
da
tabela
circuitos.

circuitos=

{circuitId,circuitRef,name,location,country,lat,lng,alt,url}

```
CREATE TABLE circuitos (  
    circuitId DECIMAL(4) NOT NULL PRIMARY KEY,  
    circuitRef VARCHAR(20) NOT NULL,  
    name VARCHAR(60) NOT NULL,  
    location VARCHAR(40) NOT NULL,  
    country VARCHAR(20) NOT NULL,  
    lat FLOAT (10) NOT NULL,  
    lng FLOAT (10) NOT NULL,  
    alt FLOAT (10) NOT NULL,  
    url VARCHAR(60) NOT NULL  
);
```

Criando a tabela circuitos

The screenshot displays a PostgreSQL browser window with the following components:

- Browser Panel (Left):** Shows a tree view of the database structure. Under the 'public' schema, the 'Tables (1)' folder is expanded, showing the 'circuitos' table.
- Query Editor (Right):** Contains the SQL command:

```
1 CREATE TABLE circuitos (  
2     circuitId DECIMAL(4) NOT NULL PRIMARY  
3     circuitRef VARCHAR(20) NOT NULL,  
4     name VARCHAR(60) NOT NULL,  
5     location VARCHAR(40) NOT NULL,  
6     country VARCHAR(20) NOT NULL,  
7     lat FLOAT (10) NOT NULL,  
8     lng FLOAT (10) NOT NULL,  
9     alt FLOAT (10) NOT NULL,  
10    url VARCHAR(60) NOT NULL  
11 );  
12
```
- Query History (Right):** Shows the executed query and its output:

```
Data output Messages Notifications  
CREATE TABLE  
Query returned successfully in 124 msec.
```

Através do comando `psql \copy` é possível importar dados de um arquivo para uma tabela.

COPY: comando utilizado para importar a tabela

`COPY tabela_a_ser_copiada`

nome da tabela que receberá os dados

O arquivo
pode
estar

(

X,

(...): dentro do parenteses colocamos as colunas que receberão os dados. Neste caso, utilizamos os nomes das

DELIMITER: após o delimiter devemos colocar o separador que o arquivo utiliza. O separador poder ser ";", ",", "|" entre outros. Neste caso, o separador do arquivo é ";"

FROM: após o "FROM" devemos colocar o diretório que está localizado o arquivo. O diretório sempre deve usar "/", nunca "\", tanto

em windows quanto em linux. O arquivo que

maquina
cliente.

`DELIMITER ';' ;`

`CSV HEADER;`

CSV HEADER: utilize "CSV HEADER" caso não queira importar a primeira linha. Para este arquivo não utilizamos a primeira linha, pois é o cabeçalho; caso contrário, caso não queira importar a primeira linha, pois é o cabeçalho;

```
COPY circuitos
```

```
(
```

```
    circuitId,
```

```
        circuitRef,
```

```
        name,
```

```
        location,
```

```
        country,
```

```
        lat,
```

```
        lng,
```

```
        alt,
```

```
        url,
```

```
)
```

```
FROM 'F:\2022-2\BD\Kthannus\Formula1\circuits.csv'
```

```
DELIMITER ';' 
```

```
CSV HEADER;
```



Executando o comando

pgAdmin File Object Tools Help

Browser Properties SQL formula1/postgres@PostgreSQL 14*

Publications

Schemas (1)

public

- Aggregates
- Collations
- Domains
- FTS Configurations
- FTS Dictionaries
- FTS Parsers
- FTS Templates
- Foreign Tables
- Functions
- Materialized Views
- Operators
- Procedures
- Sequences
- Tables (1)**
 - circuitos**
 - Columns
 - Constraints
 - Indexes
 - RLS Policies
 - Rules
 - Triggers

formula1/postgres@PostgreSQL 14

Query Query History

```
1 COPY circuitos
2 (
3     circuitId,
4     circuitRef,
5     name,
6     location,
7     country,
8     lat,
9     lng,
10    alt,
11    url
12 )
13 FROM 'F:\2022-2\BD\Kthannus\Formula1\circuits.csv'
14 DELIMITER ','
15 CSV HEADER;
16
```

Data output Messages Notifications

COPY 79

Query returned successfully in 110 msec.

pgAdmin 4

pgAdmin File Object Tools Help

Browser Properties SQL copy_circuitos.sql*

formula1/postgres@PostgreSQL 14

No limit

Query Query History

```

1 SELECT name, country FROM circuitos;
2
3

```

Data output Messages Notifications

	name	country
	character varying (60)	character varying (20)
1	Albert Park Grand Prix Circuit	Australia
2	Sepang International Circuit	Malaysia
3	Bahrain International Circuit	Bahrain
4	Circuit de Barcelona-Catalunya	Spain
Total rows: 79 of 79		Query complete 00:00:00.074

Browser tree:

- Publications
- Schemas (1)
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Operators
 - Procedures
 - Sequences
 - Tables (1)
 - circuitos
 - Columns (9)
 - circuitid
 - circuitref
 - name
 - location
 - country
 - lat
 - lng
 - alt

Execício

Criar e
popular as
seguintes
tabelas
tabelas:

- Racers
- Drivers
- Constructors
- ...

...DDL

SQL

DDL - Comando ALTER TABLE - Padrão ISO

Modifica tabelas já definidas

Sintaxe geral de um Comando ALTER TABLE

ALTER TABLE <nome da tabela>

ADD <definição de Coluna>

ADD <Restrição de integridade> - Chaves primária, Candidatas, Estrangeiras.

ALTER <definição de Coluna> SET DATA TYPE <Tipo>

ALTER <definição de Coluna> {SETIDROP} DEFAULT<default>

ALTER <definição de Coluna> {SETIDROP} NOT NULL

DROP <definição de Coluna>

DROP CHECK I DROP <nome da chave>

DROP UNIQUE (<definição de Coluna>, ...)

DROP PRIMARY KEY

DROP FOREIGN KEY

RENAME <novo nome> - Renomeia a tabela

RENAME <Atributo> TO <novo atributo>

onde <definição de coluna> pode ser:

<Nome Atributo> <Tipo de Dado>

{SETIDROP} [NULL] I [DEFAULT default-value]

ADD

Adicionando uma coluna.
O comando então será:

```
-- adicionar telefone VARCHAR(30)  
ALTER TABLE aluno  
    ADD telefone VARCHAR(30);
```

postgres/postgres@PostgreSQL 14

Query Query History

```
8 ALTER TABLE aluno
9     ADD telefone VARCHAR(30);
10
11
```

Data Loading... Notifications

ALTER TABLE

Query returned successfully in 60 msec.

- FTS Configurations
- FTS Dictionaries
- FTS Parsers
- FTS Templates
- Foreign Tables
- Functions
- Materialized Views
- Operators
- Procedures
- 1.3 Sequences
- Tables (3)
 - aluno
 - Columns (5)
 - nusp
 - nome
 - cidade
 - idade
 - telefone

postgres/postgres@PostgreSQL 14

Query Query History

```
8 --ALTER TABLE aluno
9 -- ADD telefone VARCHAR(30);
10 SELECT telefone FROM aluno;
11
```

Data output Messages Notifications

telefone
character varying (30)

- FTS Configurations
- FTS Dictionaries
- FTS Parsers
- FTS Templates
- Foreign Tables
- Functions
- Materialized Views
- Operators
- Procedures
- 1.3 Sequences
- Tables (3)
 - aluno
 - Columns (5)
 - nusp
 - nome
 - cidade
 - idade
 - telefone

Exemplo:
Projeto
do banco
de dados
criamos
a
seguinte
tabela

The screenshot shows a PostgreSQL client interface with the following elements:

- Properties: postgres/postgres@PostgreSQL 14*
- Connection: postgres/postgres@PostgreSQL 14
- Query: `SELECT idade FROM aluno;`
- Data: Loading...
- Result: A table with one column named 'idade' of type 'numeric (3)'.

Mas desejamos que idade seja INTEGER de 2.

DDL - Comando ALTER TABLE - Padrão ISO

Modifica tabelas já definidas

Sintaxe geral de um Comando ALTER TABLE



```
ALTER TABLE <nome da tabela>  
  ADD <definição de Coluna>  
  ADD <Restrição de integridade> - Chaves primária, Candidatas, Estrangeiras.  
  ALTER <definição de Coluna> SET DATA TYPE <Tipo>  
  ALTER <definição de Coluna> {SETIDROP} DEFAULT<default>  
  ALTER <definição de Coluna> {SETIDROP} NOT NULL  
  DROP <definição de Coluna>  
  DROP CHECK      I DROP <nome da chave>  
  DROP UNIQUE (<definição de Coluna>, ...)  
  DROP PRIMARY KEY  
  DROP FOREIGN KEY  
  RENAME <novo nome> - Renomeia a tabela  
  RENAME <Atributo> TO <novo atributo>
```

onde <definição de coluna> pode ser:

```
<Nome Atributo> <Tipo de Dado>  
{SETIDROP} [NULL] I [DEFAULT default-value ]
```

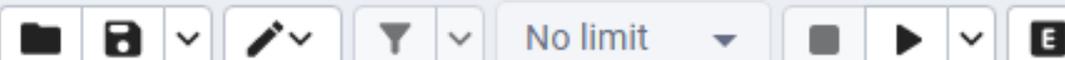
ALTER

O comando então será:

```
-- idade DECIMAL(3) para INTEGER(2)  
ALTER TABLE aluno  
    ALTER idade SET DATA TYPE INTEGER (2);
```

Properties SQL postgres/postgres@PostgreSQL 14*

postgres/postgres@PostgreSQL 14



Query Query History

```
2 ALTER TABLE aluno
3     ALTER idade SET DATA TYPE INTEGER;
4
5
```

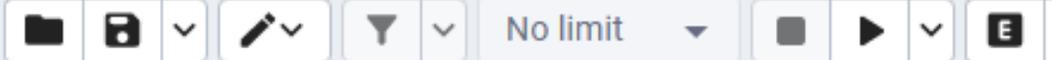
Data output Messages Notifications

ALTER TABLE

Query returned successfully in 147 msec.

Properties SQL postgres/postgres@PostgreSQL 14*

postgres/postgres@PostgreSQL 14



Query Query History

```
2 SELECT idade FROM aluno;
3
4
5
```

Data output Messages Notifications



idade	integer	🔒
-------	---------	---

DDL - Comando ALTER TABLE - Padrão ISO

Modifica tabelas já definidas

Sintaxe geral de um Comando ALTER TABLE

ALTER TABLE <nome da tabela>
ADD <definição de Coluna>
ADD <Restrição de integridade> - Chaves primária, Candidatas, Estrangeiras.
ALTER <definição de Coluna> SET DATA TYPE <Tipo>
ALTER <definição de Coluna> {SETIDROP} DEFAULT<default>
ALTER <definição de Coluna> {SETIDROP} NOT NULL
DROP <definição de Coluna>
DROP CHECK
DROP <nome da chave>
DROP UNIQUE (<definição de Coluna>, ...)
DROP PRIMARY KEY
DROP FOREIGN KEY
RENAME <novo nome> - Renomeia a tabela
RENAME <Atributo> TO <novo atributo>

onde <definição de coluna> pode ser:

<Nome Atributo> <Tipo de Dado>
{SETIDROP} [NULL] I [DEFAULT default-value]

DROP

Apagando uma coluna.
O comando então será:

```
-- apagar a coluna telefone  
ALTER TABLE aluno  
    DROP telefone;
```

Browser



Properties SQL postgres/postgres@Pos

- > FTS Configurations
- > FTS Dictionaries
- > Aa FTS Parsers
- > FTS Templates
- > Foreign Tables
- > Functions
- > Materialized Views
- > Operators
- > Procedures
- > 1.3 Sequences
- ▼ Tables (3)
 - ▼ aluno
 - ▼ Columns (4)
 - nusp
 - nome
 - cidade
 - idade
 - > Constraints

postgres/postgres@PostgreSQL 14

Folder Save Filter No limit

Query Query History

```
8 ALTER TABLE aluno
9         DROP telefone;
10 -- refresh
11
```

Data output Messages Notifications

ALTER TABLE

Query returned successfully in 59 msec.

- ▼ Tables (1)
 - ▼ circuitos
 - Columns (9)
 - circuitid
 - circuitref
 - name
 - location
 - country
 - lat
 - lng
 - alt
 - url

- ▶ Constraints
- ▶ Indexes
- ▶ RLS Policies
- ▶ Rules
- ▶ Triggers
- ▶ Trigger Functions
- ▶ Types
- ▶ Views

- ▶ Subscriptions
- ▶ postgres
- ▶ tes01
- ▶ teste
- ▶ usp

formula1/postgres@PostgreSQL 14










Query Query History

1 SELECT * FROM circuitos;

Data output Messages Notifications








	circuitid [PK] numeric (4)	circuitref character varying (20)	name character varying (60)	location character varying (40)	country character varying (20)	lat real	lng real	alt real
1	1	albert_park	Albert Park Grand Pri...	Melbourne	Australia	-37.8497	144.968	
2	2	sepang	Sepang International ...	Kuala Lumpur	Malaysia	2.76083	101.738	
3	3	bahrain	Bahrain International ...	Sakhir	Bahrain	26.0325	50.5106	
4	4	catalunya	Circuit de Barcelona-...	Montmeló	Spain	41.57	2.26111	
5	5	istanbul	Istanbul Park	Istanbul	Turkey	40.9517	29.405	
6	6	monaco	Circuit de Monaco	Monte-Carlo	Monaco	43.7347	7.42056	
7	7	villeneuve	Circuit Gilles Villeneuve	Montreal	Canada	45.5	-73.5228	
8	8	magny_cours	Circuit de Nevers Ma...	Magny Cours	France	46.8642	3.16361	
9	9	silverstone	Silverstone Circuit	Silverstone	UK	52.0786	-1.01694	



DDL - Comando ALTER TABLE - Padrão ISO

Modifica tabelas já definidas

Sintaxe geral de um Comando ALTER TABLE

```
ALTER TABLE <nome da tabela>  
  ADD <definição de Coluna>  
  ADD <Restrição de integridade> - Chaves primária, Candidatas, Estrangeiras.  
  ALTER <definição de Coluna> SET DATA TYPE <Tipo>  
  ALTER <definição de Coluna> {SETIDROP} DEFAULT<default>  
  ALTER <definição de Coluna> {SETIDROP} NOT NULL  
  DROP <definição de Coluna>  
  DROP CHECK  
  DROP <nome da chave>  
  DROP UNIQUE (<definição de Coluna>, ...)  
  DROP PRIMARY KEY  
  DROP FOREIGN KEY  
  RENAME <novo nome> - Renomeia a tabela  
  RENAME <Atributo> TO <novo atributo>
```

onde <definição de coluna> pode ser:

```
<Nome Atributo> <Tipo de Dado>  
{SETIDROP} [NULL] I [DEFAULT default-value ]
```

RENAME

Mudando o nome de uma coluna.
O comando então será:

```
-- mudar a coluna cidade para cidades  
ALTER TABLE aluno  
    RENAME cidade TO cidades;
```

Browser postgres/postgres@PostgreSQL 14*

- > FTS Configurations
- > FTS Dictionaries
- > Aa FTS Parsers
- > FTS Templates
- > Foreign Tables
- > Functions
- > Materialized Views
- > Operators
- > Procedures
- > 1.3 Sequences
- ▼ Tables (3)
 - ▼ aluno
 - ▼ Columns (4)
 - nusp
 - nome
 - cidade
 - idade
 - > Constraints

postgres/postgres@PostgreSQL 14

Query Query History

```
1 -- mudar a coluna cidade para cidades
2 ALTER TABLE aluno
3     RENAME cidade TO cidades;
4
```

Data output Messages Notifications

ALTER TABLE

Browser postgres/postgres@PostgreSQL 14

- > FTS Configurations
- > FTS Dictionaries
- > Aa FTS Parsers
- > FTS Templates
- > Foreign Tables
- > Functions
- > Materialized Views
- > Operators
- > Procedures
- > 1.3 Sequences
- ▼ Tables (3)
 - ▼ aluno
 - ▼ Columns (4)
 - nusp
 - nome
 - cidades
 - idade
 - > Constraints

postgres/postgres@PostgreSQL 14

Query Query History

```
1 -- mudar a coluna cidade para cida
2 ALTER TABLE aluno
3     RENAME cidade TO cidades;
4
```

Data output Messages Notifications

ALTER TABLE

Query returned successfully in 123 msec.

Exercícios: faça as seguintes querrys:

```
ALTER TABLE Professor  
ADD CorCabelos CHAR(25) DEFAULT 'Branco';
```

```
ALTER TABLE Aluno  
ADD Altura INT DEFAULT NULL;
```

```
ALTER TABLE Aluno  
DROP Altura;
```

```
ALTER TABLE Professor  
ALTER CorCabelos CHAR(30) ;
```

```
ALTER TABLE Aluno  
DROP MonitoraDiscip;
```

DDL - Comando DROP TABLE

Remove completamente uma tabela e sua definição

Sintaxe geral de um Comando DROP TABLE

```
DROP TABLE [IF EXISTS] <nome da tabela> [, ...]  
[CASCADE | RESTRICT];
```

CASCADE: todas as visões e restrições (constraints) que referenciam o atributo são removidas automaticamente

RESTRICT: O atributo só é removido se não houver nenhuma visão ou restrição que o referencie

DROP TABLE

Exemplo:

```
-- apagar a tabela aluno  
DROP TABLE aluno; ;
```



- > Operators
- > Procedures
- > 1.3 Sequences
- ▼ Tables (2)
 - ▼ disciplina
 - > Columns
 - > Constraints
 - > Indexes
 - > RLS Policies
 - > Rules
 - > Triggers
 - ▼ professor
 - > Columns
 - > Constraints
 - > Indexes
 - > RLS Policies
 - > Rules
 - > Triggers
 - > Trigger Functions
 - > Types
 - > Views
- > Subscriptions
- > tes01

postgres/postgres@PostgreSQL 14



Query Query History

```
1 DROP TABLE Aluno;  
2  
3
```

Data output Messages Notifications

DROP TABLE

Query returned successfully in 128 msec.

