



**Ministério da Educação  
Instituto Federal de Mato Grosso  
Campus Cuiabá – Cel. Octayde Jorge da Silva  
Departamento de Área de Informática**



# **Aula RED 11 – PostgreSQL: Comandos SQL (Fundamentos de BD)**

**Profª Juliana Saragiotto Silva**

IFMT - Campus Cuiabá

Cuiabá, MT – 08/03/2021



# Agenda

1. Objetivo da aula
2. PostgreSQL: apresentação do ambiente
3. Comandos SQL:
  - 3.1. Insert (inserção de dados)
  - 3.2. Select (consulta de dados)
4. Atividade da semana



# 1. Objetivo da aula

Apresentar o ambiente de trabalho do PostgreSQL  
(ferramenta PGAdmin) e os comandos básicos  
de Linguagem de Consulta Estruturada  
(SQL - *Structured Query Language*).



## 2. PostgreSQL: PGAdmin

← → ↻ ⓘ 127.0.0.1:35203/browser/

**pgAdmin** File ▾ Object ▾ Tools ▾ Help ▾

Browser

- Servers (1)
  - PostgreSQL 13
    - Databases (2)
      - PROTOTIPO**
        - Cast
        - Catalogs
        - Event Triggers
        - Extensions
        - Foreign Data Wrappers
        - Languages
        - Schemas (1)

→

- Schemas (1)
  - public**
    - Collations
    - Domains
    - FTS Configurations
    - FTS Dictionaries
    - FTS Parsers
    - FTS Templates
    - Foreign Tables
    - Functions
    - Materialized Views
    - Procedures
    - Sequences
    - Tables (2)**
      - departamento
      - funcionario
    - Trigger Functions



## 2. PostgreSQL: Tools -> Query Tool

**pgAdmin** File ▾ Object ▾ Tools ▾ Help ▾

Browser

- ▾ Servers (1)
  - ▾ PostgreSQL 13
    - ▾ Databases (2)

Tools ▾

- New ERD Project (Beta)
- Query Tool**
- Schema Diff

Dashboard Properties SQL Statistics Dependencies Dependents [PROTOTIPO/postgres@Postgre](#)

Query Editor Query History

1

Notifications

Recorded time	Event	Process ID	Payload
No data found			



## 2. PostgreSQL: Tools -> Query Tool

Dashboard Properties SQL Statistics Dependencies Dependents **PROTOTIPO/postgres@PostgreSQL 13 \*** X

Icons: [Database] [Folder] [Save] [Grid] [Search] [Filter] [Copy] [Paste] [Delete] [Edit] [Filter] [Limit: No limit] [Play] [Hand] [List] [Refresh] [Refresh]

Query Editor Query History Scratch Pad X

1 **SELECT** \*  
2 **FROM** departamento

**F5**

Data Output Explain Messages

	id_departamento [PK] integer	nome character varying (100)	sigla character varying (15)	ramal character varying (11)	responsavel character varying (100)



### 3. Comandos SQL: INSERT

Sintaxe:

**INSERT into Nome\_tabela**

**(atributo1, atributo2, .., atributo n)**

**VALUES**

**(1, 'Nelcael', ..., '08/03/2021')**



### 3. Comandos SQL: INSERT

Exemplo (Departamento):

**INSERT** into **Departamento**

(id\_departamento, nome, sigla, ramal, responsavel)

**VALUES**

(1, 'Tecnologia da Informação', 'TI', '1402', 'Gabriel');





### 3. Comandos SQL: SELECT

Query Editor

Query History

1

SELECT \*

2

FROM departamento

Data Output

Explain

Messages

	<div>id_departamento</div> <div>[PK] integer</div>	<div>nome</div> <div>character varying (100)</div>	<div>sigla</div> <div>character varying (15)</div>	<div>ramal</div> <div>character varying (11)</div>	<div>responsavel</div> <div>character varying (100)</div>
1	1	Tecnologia da Informação	TI	1402	Gabriel



### 3. Comandos SQL: INSERT (bloco)

Exemplo (Departamento):

**INSERT into Departamento**

**(id\_departamento, nome, sigla, ramal, responsavel)**

**VALUES**

**(2, 'Gestão de Pessoas', 'GP', '1403', 'Keila'),**

**(3, 'Financeiro', 'FN', '1404', 'Renan'),**

**(4, 'Comercial', 'CM', '1405', 'Lucas');**



## 3. Comandos SQL: SELECT

Query Editor

Query History

Scratch Pad

1 **SELECT** \*

2 **FROM** Departamento

Data Output

Explain

Messages

	id_departamento [PK] integer	nome character varying (100)	sigla character varying (15)	ramal character varying (11)	responsavel character varying (100)
1	1	Tecnologia da Informação	TI	1402	Gabriel
2	2	Gestão de Pessoas	GP	1403	Keila
3	3	Financeiro	FN	1404	Renan
4	4	Comercial	CM	1405	Lucas



### 3. Comandos SQL: INSERT

Exemplo (Funcionario):

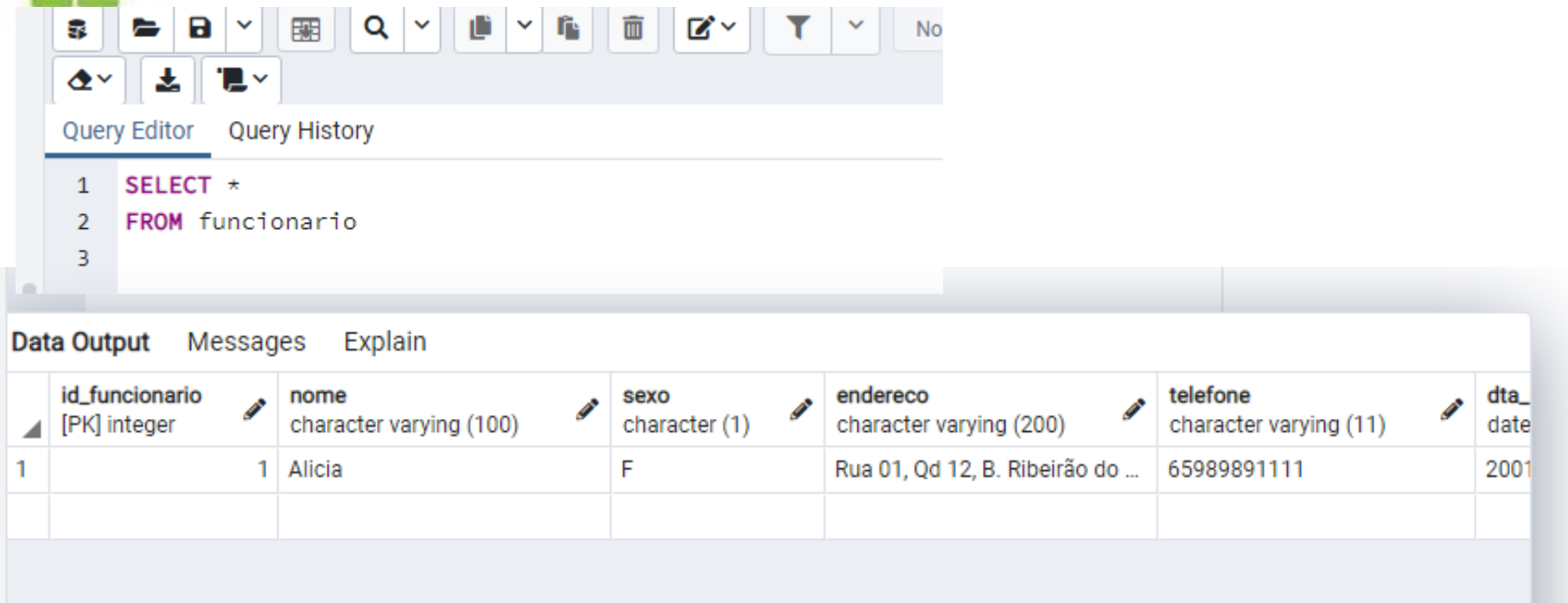
**INSERT** into **Funcionario**

(id\_funcionario, nome, sexo, endereco,  
telefone, dta\_nasc, email,  
numero\_matricula, dta\_admissao, id\_departamento)

**VALUES**

(1, 'Alicia', 'F', 'Rua 01, Qd 12, B. Ribeirão do Lipa',  
'65989891111', '23/10/2001', 'alicia@gmail.com',  
2001, '02/01/2018', 5);

### 3. Comandos SQL: SELECT



The screenshot displays a database management interface. At the top, a toolbar contains icons for file operations, search, and filtering. Below the toolbar, the 'Query Editor' tab is active, showing a SQL query:

```
1 SELECT *  
2 FROM funcionario  
3
```

Below the query editor, the 'Data Output' tab is selected, displaying the results of the query in a table. The table has seven columns: **id\_funcionario** (PK integer), **nome** (character varying (100)), **sexo** (character (1)), **endereco** (character varying (200)), **telefone** (character varying (11)), and **dta\_** (date). The first row of data shows the record for Alicia.

	id_funcionario [PK] integer	nome character varying (100)	sexo character (1)	endereco character varying (200)	telefone character varying (11)	dta_ date
1	1	Alicia	F	Rua 01, Qd 12, B. Ribeirão do ...	65989891111	2001



### 3. Comandos SQL: INSERT (bloco)

Exemplo (Funcionario):

#### INSERT into **Funcionario**

(id\_funcionario, nome, sexo, endereco, telefone,  
dta\_nasc, email, numero\_matricula, dta\_admissao, id\_departamento)

#### VALUES

(2, 'Jessica', 'F', 'Rua 20, Qd 5, B. Morada do Ouro', '65999892222',  
'08/03/2002', 'jessica@gmail.com', 2002, '02/01/2019', 1),  
(3, 'João Marcos', 'M', 'Rua X, B. CPA', '65988982525', '07/01/2001',  
'joaomarcos@gmail.com', 2003, '02/01/2019', 2),  
(4, 'Lucas Adriano', 'M', 'Rua Z, Qd Y, B. Centro', '65998980202',  
'26/04/1995', 'lucasadriano@gmail.com', 2004, '30/01/2015', 2),  
(5, 'William', 'M', 'Rua A, Qd 10, B. Cachorro Pelado', '65997970404',  
'08/03/1998', 'william@gmail.com', 2005, '08/03/2015', 3);



Query Editor    Query History

```
1 select *
2 from funcionario
```

Data Output Messages

[illegible]



### 3. Comandos SQL: SELECT (com mais de 1 tabela)

Sintaxe:

**SELECT** atributo\_x, atributo\_y, atributo\_z -- atributos que você  
necessita trazer na consulta

**FROM** Tabela\_1, Tabela\_2 -- pode ter mais de 2 tabelas

**WHERE** Tabela\_1.atributo\_z = Tabela\_2.atributo\_z

-- atributo que tem em comum nas 2 tabelas (garantir a integridade)

**ORDER BY** 1, 2 – ordenar a consulta SQL, pelos 2 primeiros atributos





### 3. Comandos SQL: SELECT (com mais de 1 tabela)

Exemplo:

```
SELECT Departamento.Nome, Funcionario.Nome,  
        Email, Telefone  
  
FROM Departamento, Funcionario  
  
WHERE Departamento.Id_Departamento = Funcionario.Id_Departamento  
  
ORDER BY 1, 2
```



### 3. Comandos SQL: SELECT (com mais de 1 tabela)

PROTÓTIPO/postgres@PostgreSQL 13 ▾

Query Editor   Query History

```
1 SELECT Departamento.Nome, Funcionario.Nome,  
2     Email, Telefone  
3 FROM Departamento, Funcionario  
4 WHERE Departamento.Id_Departamento = Funcionario.Id_Departamento  
5 ORDER BY 1, 2  
6
```

Data Output   Messages

	nome character varying (100)	nome character varying (100)	email character varying (50)	telefone character varying (11)
1	Financeiro	William	william@gmail.com	65997970404
2	Gestão de Pessoas	João Marcos	joaomarcos@gmail.com	65988982525
3	Gestão de Pessoas	Lucas Adriano	lucasadriano@gmail.com	65998980202
4	Tecnologia da Informação	Jessica	jessica@gmail.com	65999892222



## 4. Atividade da semana

Realizar o povoamento para as tabelas do cenário Clinica\_Medica, a partir da seguinte orientação:

- 3 médicos;
- 6 pacientes;
- 12 consultas, sendo 02 para cada paciente, alternando os médicos

Obs.: Para o cenário 1 (Funcionário Departamento) foi pedido o povoamento de 4 departamentos e 8 funcionários (sendo 2 de cada departamento).



**Obrigada!**

**Juliana Saragiotto Silva**

**Email:** [juliana.silva@cba.ifmt.edu.br](mailto:juliana.silva@cba.ifmt.edu.br)