

Database Integration

Prof. Jonilson Sousa

Mastering Python

Aula 01

- O que é banco de dados;
- Relacional x Não Relacional;
- Modelagem de dados;
- Linguagem SQL;
- DDL - CREATE.

O que é um banco de dados?

O que é um banco de dados?

- É um conjunto lógico e ordenado de dados ou informações armazenados que possuem algum significado;
- Ele pode conter qualquer tipo de dados;
- E podem ser de qualquer tamanho.
- **Sistema de Gerenciamento de Banco de Dados (SGBD):** É responsável por controlar (gerenciar) os bancos de dados;


Alguns tipos de banco de dados








- **Bancos de dados relacionais:** Os dados são organizados como um conjunto de tabelas, com colunas e linhas, assim temos as informações estruturadas;
- **Bancos de dados NoSQL:** Foi criado para ser de alto desempenho, e escalabilidade horizontal, e os dados são armazenados como documentos, chave-valor, em colunas, geralmente usando o padrão JSON;
- **Bancos de dados orientados a objetos:** Os dados são representados na forma de objetos;
- **Bancos de dados gráficos:** Usa a teoria dos grafos para armazenar, mapear e consultar os relacionamentos.

Relacional

- Um banco de dados que armazena e fornece acesso a pontos de dados relacionados entre si;
- Organização de dados em **tabelas, com linhas e colunas**, ou seja:
 - Cada tabela representa uma **ENTIDADE**;
 - Cada tabela pode ter um ou mais **ATRIBUTOS** que são as colunas;
 - Cada linha na tabela tem um registro exclusivo chamado de **CHAVE PRIMÁRIA**;
 - E cada tabela (**ENTIDADE**) pode se relacionar com outras tabelas usando as **CHAVES ESTRANGEIRAS**.

Relacional

	 id ÷	 nome ÷
1	1	Comédia
2	2	Aventura
3	3	Romance

	 id ÷	 titulo ÷	 isbn ÷	 paginas ÷	 codigo ÷	 ano ÷	 categoria_id ÷
1	1	Teto para dois	955-2891292	400	12AB	2019	3
2	2	Jogo do amor ódio	955-7445429	400	14AC	2020	1
3	3	Eu e esse meu coração	955-7121469	424	17BD	2019	3

Não relacional, NoSQL

- NoSQL? Originalmente foi chamado de *open source nonrelational database* (1998) porém anos depois (2009) voltou ao foco com o nome de NoSQL, já que não utilizam SQL para manipulação dos dados.
- Idealizado como um movimento totalmente distinto do modelo relacional;
- Possui algumas características que o faz ser amplamente utilizado no mercado:
 - **Escalabilidade horizontal:** Esse tipo de banco de dados não têm bloqueios, e isso permite a escalabilidade horizontal, e talvez essa seja o grande motivo de ser altamente utilizados no mercado atualmente;
 - **Sem estrutura ou estrutura flexível:** Não tem uma estrutura fixa como os bancos de dados relacionais, o que facilita a escalabilidade e disponibilidade.

Não relacional, NoSQL



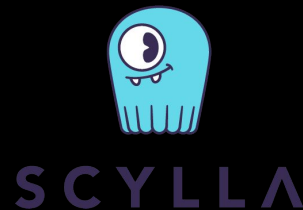
```
1 [  
2   { "_id" : 3, "nome" : "Romance" },  
3   { "_id" : 2, "nome" : "Aventura" },  
4   { "_id" : 1, "nome" : "Comédia"}  
5 ]
```



```
1 [  
2   { "_id" : 1, "titulo" : "Teto para dois", "isbn" : "955-2891292", "paginas" : 400, "codigo" : "12AB", "ano" : 2019, "categoria_id" : 3 },  
3   { "_id" : 2, "titulo" : "Jogo do amor ódio", "isbn" : "955-7445429", "paginas" : 400, "codigo" : "14AC", "ano" : 2020, "categoria_id" : 1 },  
4   { "_id" : 3, "titulo" : "Eu e esse meu coração", "isbn" : "955-7121469", "paginas" : 424, "codigo" : "17BD", "ano" : 2019, "categoria_id" : 3 }  
5 ]
```



Relacional x Não Relacional



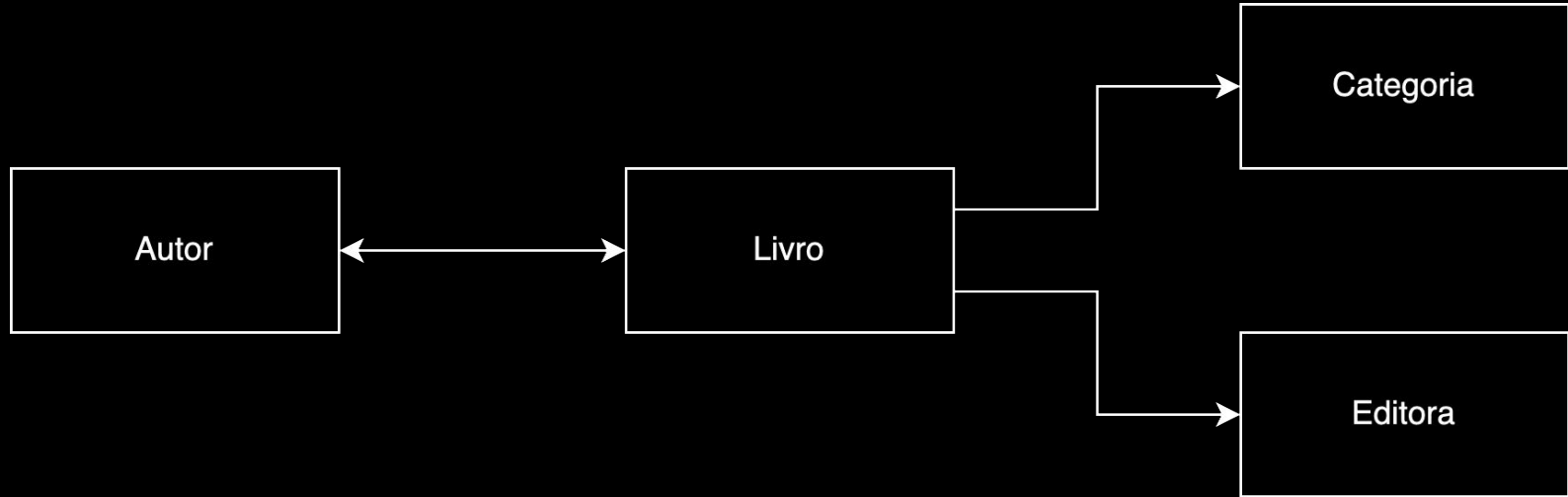
Modelagem de dados

O que é modelagem de dados?

- É o processo de criar uma representação visual, ou esquemas de dados;
- É importante para todas as partes interessadas;
- Proporciona benefícios, como:
 - Redução de erros no desenvolvimento de softwares;
 - Agiliza a criação dos bancos de dados;
 - Serve de base e documentação para todos os interessados;
 - Facilita a interação entre times.

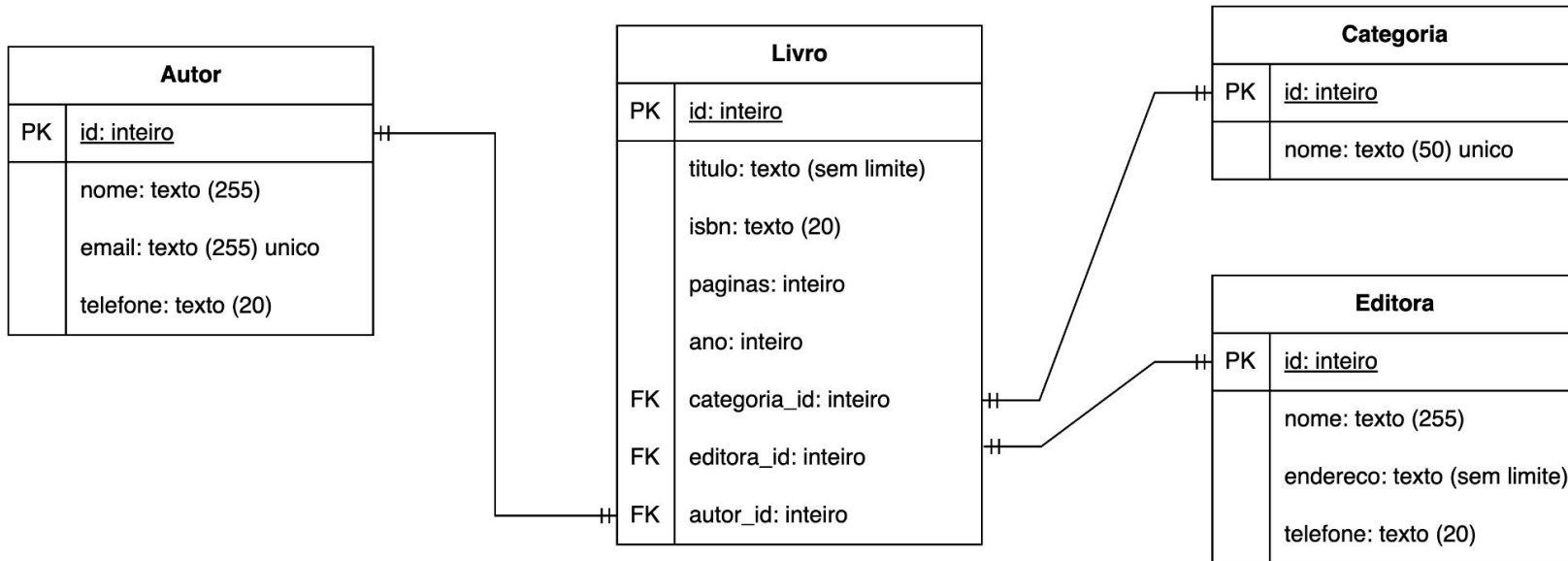
Tipos de modelagem de dados

- **Modelo de dados conceitual**, também chamados de modelo de domínio:

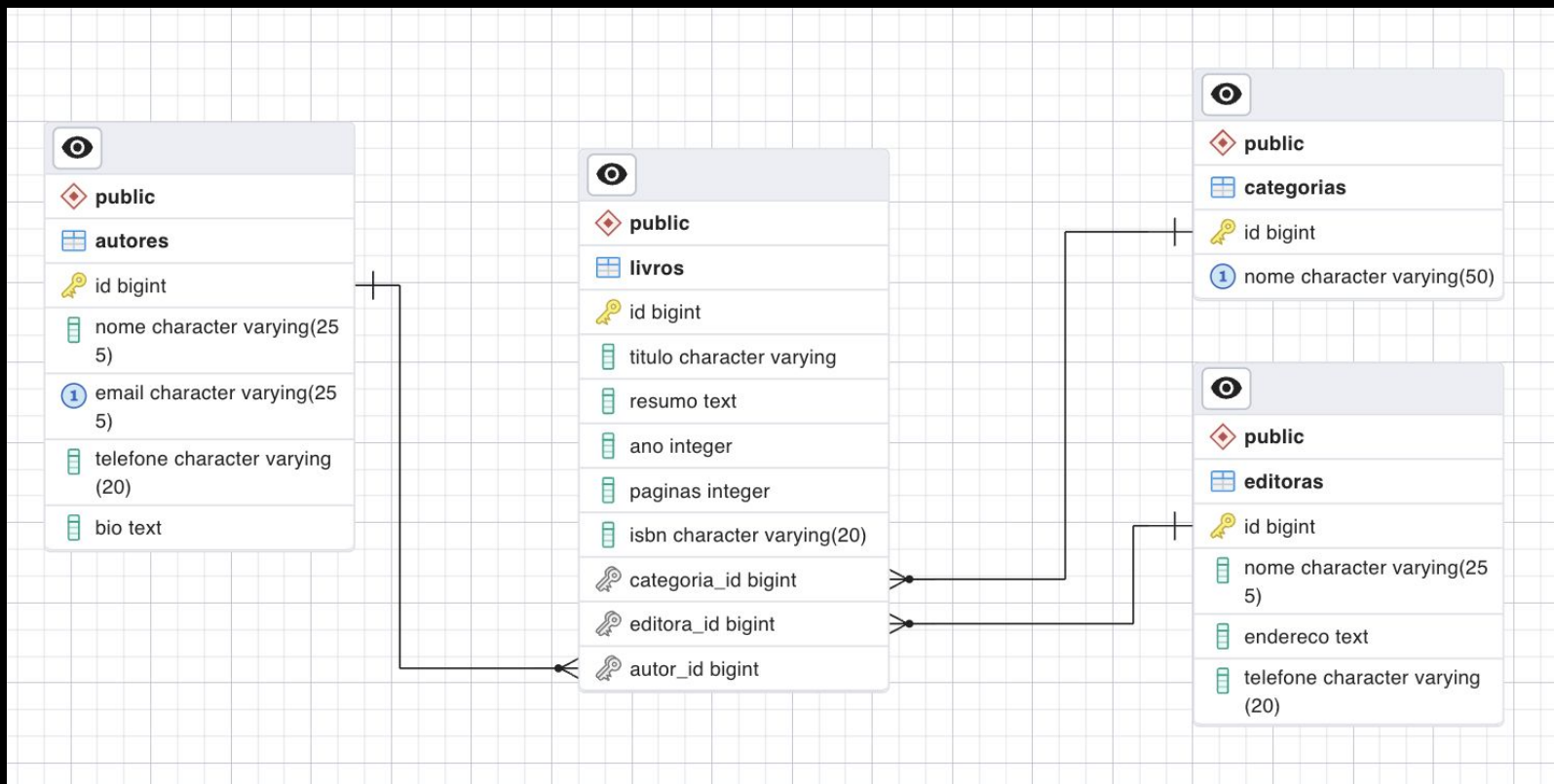


Tipos de modelagem de dados?

- Modelo de dados lógico:



Nosso modelo de dados físico



Linguagem SQL

SQL: Structured Query Language

- O que é SQL?
 - Linguagem de consulta estruturada;
 - Linguagem de programação para manipulação de dados;
 - É uma linguagem declarativa e não necessita de conhecimentos avançados para começar a usar.

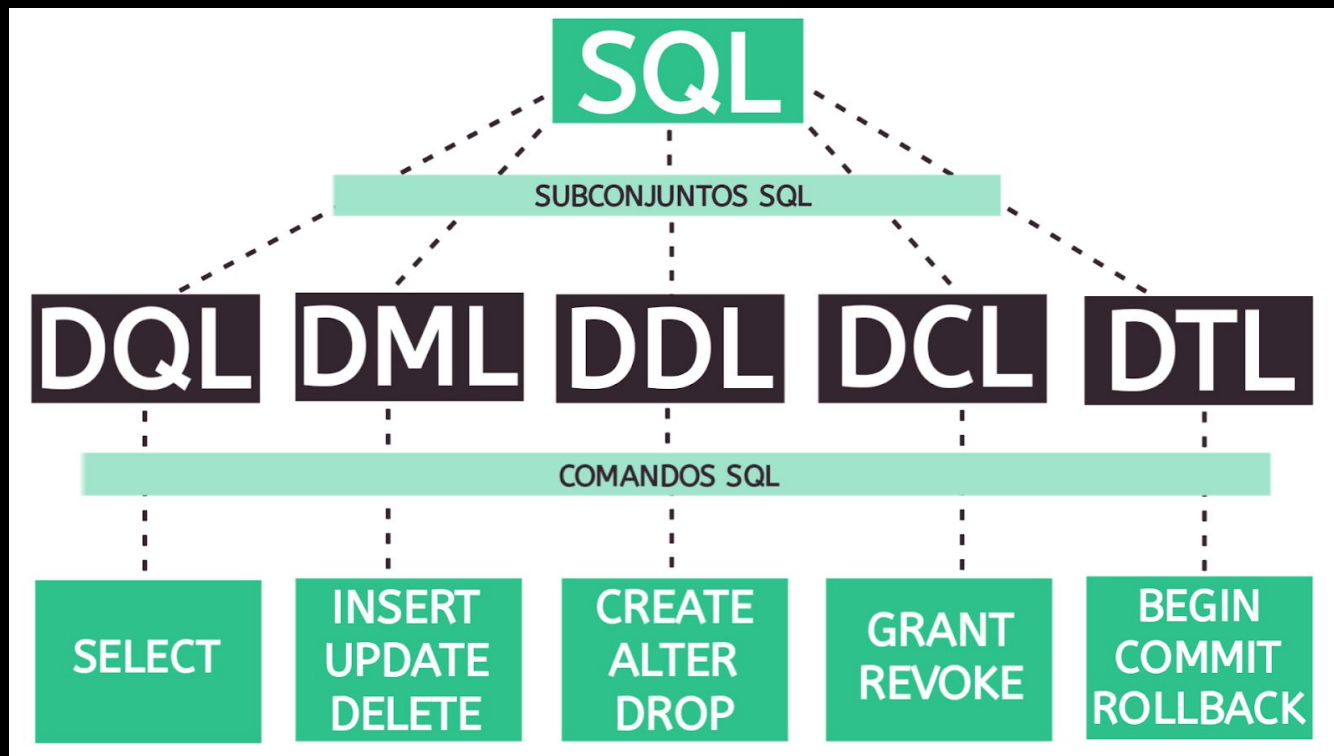
SQL: Structured Query Language

- Porque é importante aprender SQL?
 - É a linguagem padrão para banco de dados relacionais, que são amplamente utilizados;
 - É muito utilizada em aplicações dos mais diversos tipos;
 - Se integra bem com diversas linguagens de programação.
- **Importante:** Os principais banco de dados relacionais do mercado usam SQL como padrão, mas cada um tem suas características e detalhes;

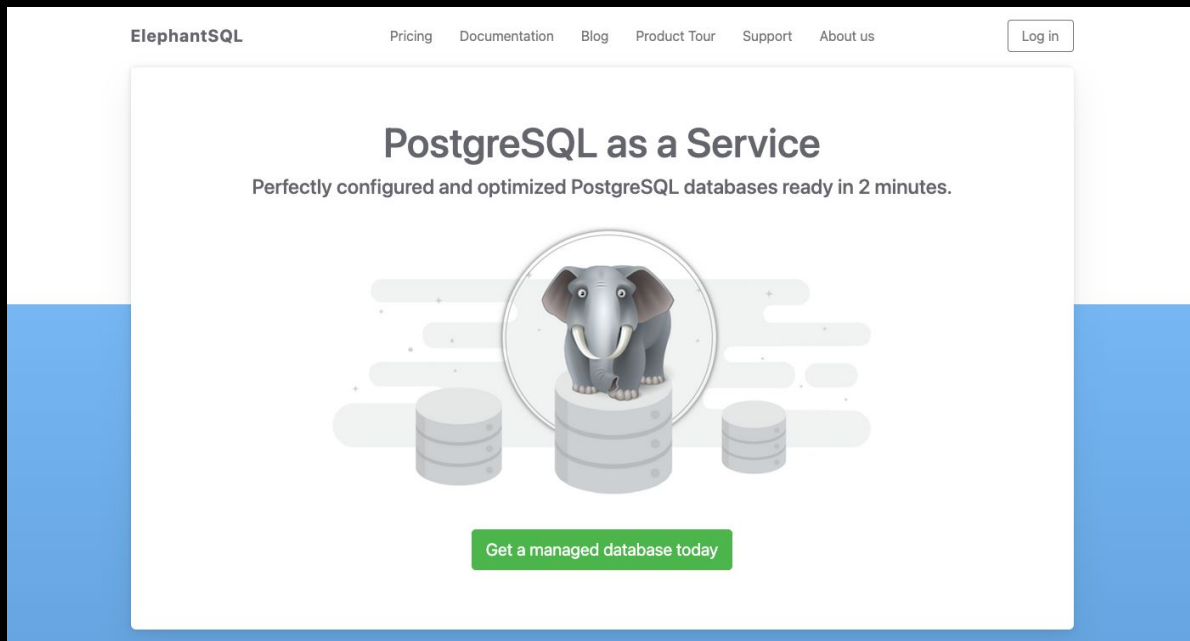
SQL: Os principais tipos de dados

- Numéricos:
 - INT
 - BIGINT
 - FLOAT
 - DOUBLE
- Data e hora:
 - DATE
 - DATETIME
 - TIMESTAMP
- Texto:
 - VARCHAR
 - CHAR
 - TEXT

Subconjuntos da linguagem SQL



Direto ao ponto com ElephantSQL



<https://www.elephantsql.com>

DDL: Data Definition Language


DDL: CREATE

CREATE DATABASE



```
1 -- Criando um banco de dados
2
3 CREATE DATABASE nome_do_banco_de_dados;
```


CREATE TABLE



```
1 -- Criando uma tabela
2
3 CREATE TABLE nome_da_tabela (
4     coluna_1 TIPO_DE_DADO PRIMARY KEY,
5     coluna_2 TIPO_DE_DADO,
6     ...
7 );
```

CREATE TABLE




```
1 -- Criando uma tabela com ID com incremento automático `auto_increment`  
2  
3 CREATE TABLE nome_da_tabela (  
4     id SERIAL PRIMARY KEY,  
5     coluna_2 TIPO_DE_DADO,  
6 );
```

CREATE TABLE

```
1 -- O que o SERIAL fazem por baixo dos panos
2
3 CREATE SEQUENCE nome_da_tabela_id_seq;
4
5 CREATE TABLE nome_da_tabela (
6     id integer NOT NULL DEFAULT nextval('nome_da_tabela_id_seq')
7 );
8
9 ALTER SEQUENCE nome_da_tabela_id_seq
10 OWNED BY nome_da_tabela.id;
11
12 -- Se for para um banco de produção use o BIGSERIAL que vai usar o `bigint`
```

CREATE TABLE



```
1 -- Criando tabela usando o "novo" GENERATED AS IDENTITY no lugar do SERIAL
2
3 CREATE TABLE nome_da_tabela (
4     id bigint GENERATED ALWAYS AS IDENTITY,
5     coluna_2 TIPO_DE_DADO,
6 );
```

CREATE TABLE

```
1 -- Criando uma tabela com chave composta
2
3 CREATE TABLE TABLE (
4     coluna_1 TIPO_DE_DADO,
5     coluna_2 TIPO_DE_DADO,
6     ...
7     PRIMARY KEY (coluna_1, coluna_2)
8 );
```

CREATE TABLE



```
1 -- Criando uma tabela com restrição UNIQUE para coluna
2
3 CREATE TABLE nome_da_tabela (
4     coluna_1 TIPO_DE_DADO PRIMARY KEY,
5     coluna_2 TIPO_DE_DADO UNIQUE
6 );
```

CREATE TABLE



```
1 -- Criando uma tabela com restrição UNIQUE para coluna como restrição de
  tabela
2
3 CREATE TABLE nome_da_tabela (
4     coluna_1 TIPO_DE_DADO PRIMARY KEY,
5     coluna_2 TIPO_DE_DADO,
6     UNIQUE (coluna_2)
7 );
```

CREATE TABLE

```
1 -- Criando tabela com chave estrangeira
2
3 CREATE TABLE nome_da_tabela (
4     coluna_1 TIPO_DE_DADO PRIMARY KEY,
5     coluna_2 TIPO_DE_DADO,
6     coluna_3 TIPO_DE_DADO,
7     ...
8     CONSTRAINT fk_nome_outra_tabela
9         FOREIGN KEY(coluna_3)
10         REFERENCES nome_outra_tabela(id)
11 );
```


Anexo

Tamanho de tipos SERIALS no PostgreSQL

Name	Storage Size	Range
SMALLSERIAL	2 bytes	1 to 32,767
SERIAL	4 bytes	1 to 2,147,483,647
BIGSERIAL	8 bytes	1 to 9,223,372,036,854,775,807