



# Fundamentos de SQL

# Quem sou?

## Alex Souza

- Master in IA
- Data Analyst
- Teacher

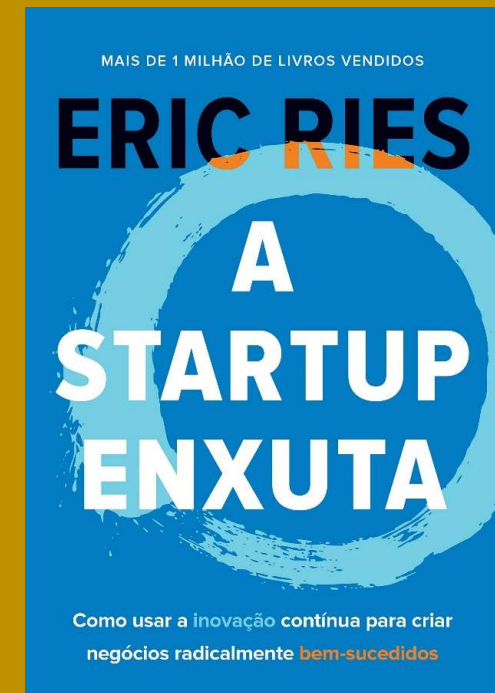
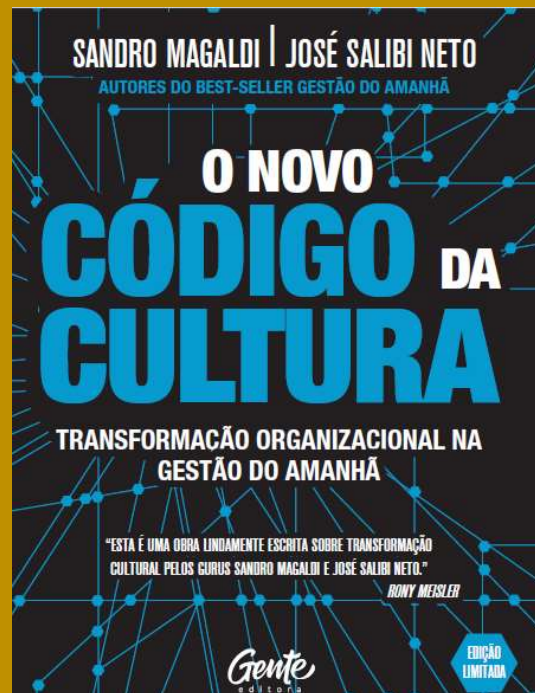


# Dicas iniciais ;)

- *Soft Skills*
- Dados
- Estudioso | Curioso
- “Resolvedor” de problemas
- Entenda o negócio / a necessidade
- Portfólio
- Inglês
- Não seja um pato
- Seja um camaleão

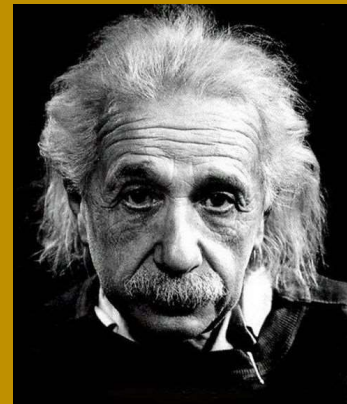


# Livros



## Pergunta certa

*“Se eu tivesse uma hora para resolver um problema e minha vida dependesse da solução, eu gastaria os primeiros 55 minutos determinando a pergunta certa a se fazer, e uma vez que eu soubesse a pergunta, eu poderia resolver o problema em menos de 5 minutos.” Albert Einstein*



# Compartilhar ;)

"A troca de conhecimento é um atalho para o aprendizado."

Alex Souza



# Robô dados

Olá, me pergunte sobre dados...



# GPDI Discord

Dicas sobre dados...





# Objetivos na disciplina



# Metodologia CRISP DM





# SISTEMAS DE BANCO DE DADOS

 FREE eBook

LEARNING  
SQL

Free unaffiliated eBook created from  
Stack Overflow contributors.

#sql

7ª edição

SISTEMAS DE  
BANCO DE DADOS

7ª edição

ELMASRI • NAVATHE

 Pearson

11

# Projeto



# Introdução



# Dados



- **Dados** podem consistir em **números, letras, palavras, imagens, gravações de voz** e assim por diante
- Os **dados** são muitas vezes vistos como o **nível mais fundamental de abstração** de onde pode-se derivar **informações** e, então, **conhecimento**
- **Conjunto de dados**: uma coleção de fatos geralmente obtida como resultado de experimentos, observações, transações ou experiências



# Múltiplas fontes de dados



# Classificação dos dados



- Dados **estruturados**:
  - Bem organizados para serem processados por computador
  - SQL - Bancos de Dados Relacionais\*
- Dados **não estruturados**:
  - Como os textos criados por humanos (documentos, e-mails) e, portanto, não preparados para serem compreendidos/consumidos por computadores
  - NOSQL - Bancos de Dados Não Relacionais
- Dados **semiestruturados**:
  - JSON (lineares), PARQUET e ORC (colunares), XML, HTML, etc.



# Estruturados



## Structured

Conforms to a  
schema



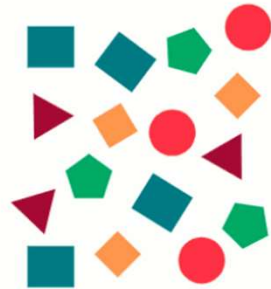
Order	CustID	Month	Item	Color	Price
101	20051	Dec	Pen	Red	2.99
102	20045	Mar	Pencil	Blue Yellow Red	3.99
103	29584	May	Eraser	Blue	1.25
104	29584	May	Pen	White	2.25
105	29584	May	Pencil	Blue Yellow Red	2.99
106	27485	Jan	Eraser	Blue Yellow	2.75
107	29574	Jan	Marker	Green	1.75
108	24447	Feb	Marker	Yellow Blue	7.25
109	26466	Jul	Pen	Black Red	5.25
110	27467	Jun	Pencil	Black	2.95

# Não Estruturados



## Unstructured

Does not fit neatly  
Into a schema

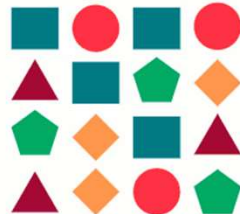


# Semi-Estruturados



## Semi-structured

Some level of organization



```
▼<div class="new-main-menu">
  ▼<div class="header-desktop-block">
    ▼<div class="container new-menu">
      ▶<a class="main-logo" rel="home" href="https://databricks.com/" title="Databricks">...</a>
    ▼<div id="new-m" class="menu-bar">
      ▼<div id="mega-menu-wrap-headerNew" class="mega-menu-wrap">
        ▶<div class="mega-menu-toggle">...</div>
        ▼<ul id="mega-menu-headerNew" class="mega-menu max-mega-menu mega-menu-horizontal" data-event="hover_intent" data-effect="fade_up" data-effect-speed="200" data-effect-mobile="disabled" data-effect-speed-mobile="0" data-panel-width="body" data-panel-inner-width="#new-m" data-mobile-force-width="false" data-second-click="close" data-document-click="collapse" data-vertical-behaviour="standard" data-breakpoint="1199" data-unbind="true">
          ▶<li class="mega-main-bar-li mega-menu-item mega-menu-item-type-custom mega-menu-item-object-custom mega-menu-item-has-children mega-menu-megamenu mega-
```

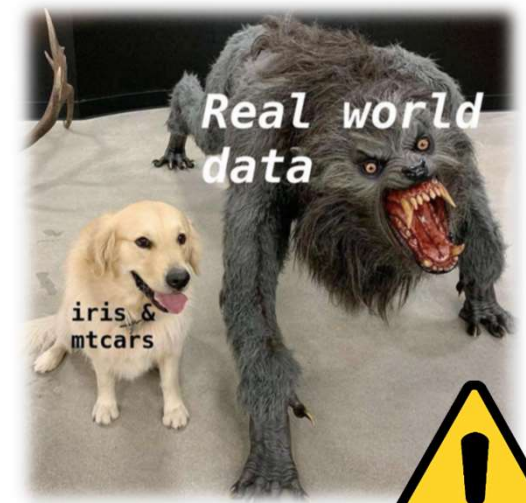
# Na vida real

(enquanto isso, na vida real...)



# Dados...

- No **mundo real**, os dados costumam se apresentar **sujos, mal-alinhados, excessivamente complexos** e **imprecisos**
  - **Não** costumam estar prontos para serem usados em tarefas de análise de dados
- É necessário preparar os dados para análise
  - **Pré-processamento de dados:**
    - **Consolidação** de dados
    - **Limpeza** de dados
    - **Transformação** de dados



## WHAT BUSINESS THINKS

We'll be done in a week!



Raw Data

Analytics

Insights



## WHAT ACTUALLY HAPPENS

After 3 months...

Raw Data



Duplicate Values

Bad Formats

No Data Dictionary

Bad Values

Analytics

Actual Insights

*This is not what we wanted!*

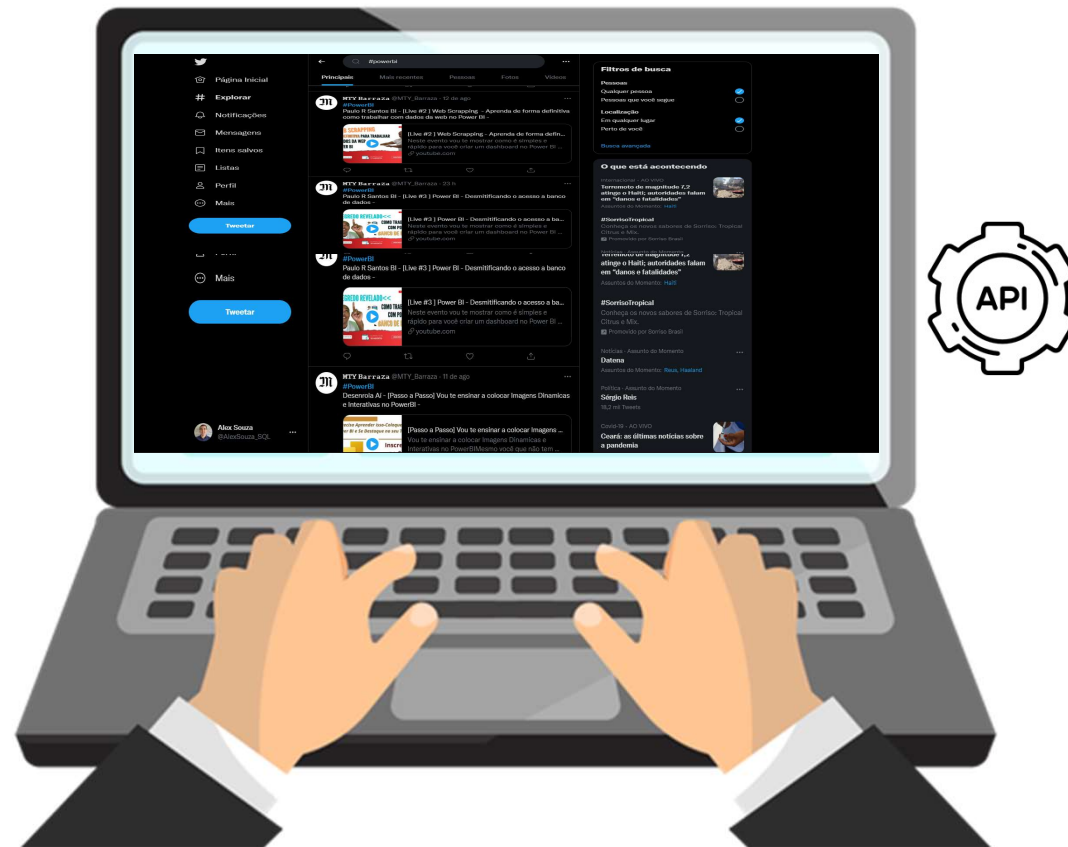
**rapidops**  
DRIVEN BY IMPACT



# Input de datos

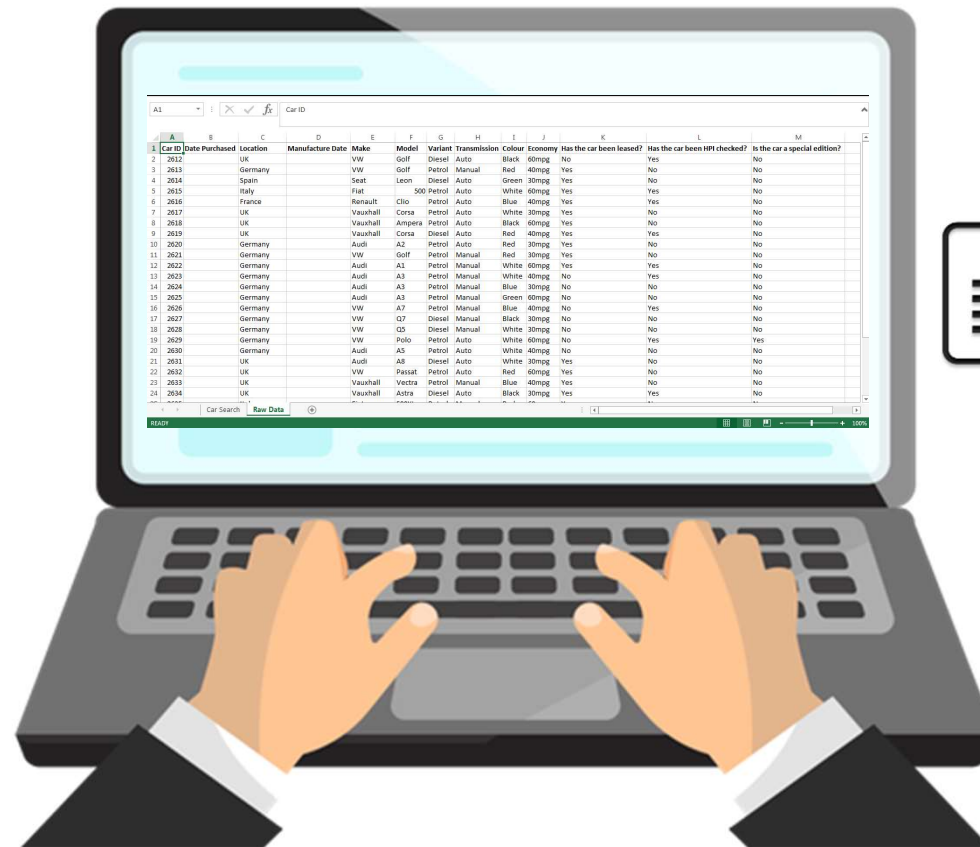
(algumas formas...)

# Inputs (Redes sociais)

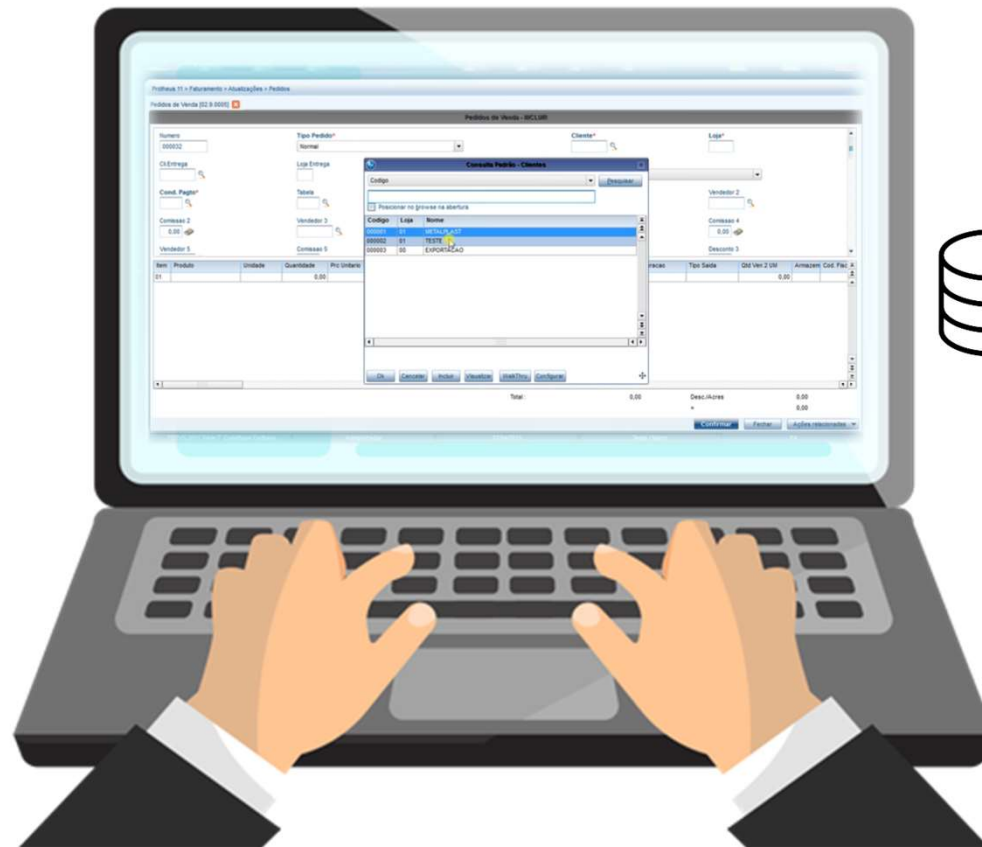




# Inputs (arquivos)



# Inputs (ERP)



# Padronização

- Camada Semântica
  - Colocar as colunas com nomes de fácil entendimento
    - Ex. CodigodaEmpresa, NomedoFornecedor...
- Padronização de Dados
  - Ao relacionar 2 fontes distintas, deixar um padrão!
    - Ex. Sexo do Funcionário
      - Em uma os dados vem como 0 (feminino) e 1 (masculino)
      - E outa os dados vem como Fem (feminino) e Mas (masculino)



# Cultura de Dados

Alinha **tecnologia**, **pessoas** e **processos** objetivando o uso **efetivo** de dados para **apoio a tomada de** decisão e estratégias organizacionais. Qualquer coisa não envolvendo estes 3 pilares, não é bem uma Cultura de dados.



# Vamos jogar?

<https://www.menti.com/ov1u5q7ows>



# Bancos de dados relacionais

*“Um **banco de dados relacional** é um tipo de **banco de dados** que armazena e fornece acesso a pontos de **dados** relacionados entre si. **Bancos de dados relacionais** são baseados no modelo **relacional**, uma maneira intuitiva e direta de representar **dados** em tabelas.” (fonte: Oracle)*

# Linguagem SQL



# Linguagem SQL

## *Structured Query Language – SQL*

- Linguagem de acesso a BD Relacionais
- Desenvolvida pela IBM – anos 70
- Inicialmente chamada de *Sequel*
- Linguagem de consulta para o sistema **R**
  - Primeiro protótipo de SBD Relacional
- Não é uma linguagem completa como java, C++, .net, python...
  - Normalmente incorporada a outra linguagem ou ferramenta para acesso a banco de dados
- ANSI-SQL é a SQL padrão, definida pelo ANSI (*American National Standard Institute*).
  - Padrão SQL ANSI e ISO:
    - SQL-86 | SQL-89 | SQL-92 | SQL:1999\*
    - SQL:2003 | SQL:2006
  - Porém, cada SGBD implementa alguns comandos específicos de SQL.
    - Exemplo: T-SQL, PL/SQL



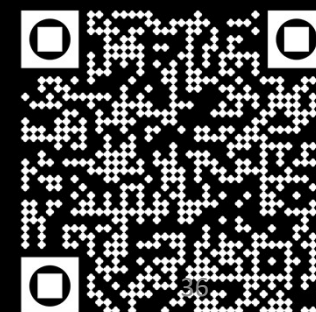
# Linguagem SQL serve para:

- **Definição de dados:** tabelas e visões (tabelas “virtuais”)
- **Recuperação de dados:** pelos usuários ou por programas
- **Manipulação de dados:** o usuário (evite ao máximo) ou programa (ERP, CRM...) pode incluir, alterar ou excluir os dados
- **Especificação de direitos de acesso:** a tabelas e visões
- **Especificação de regras para integridade** de dados
- **Especificação de bloqueios de dados para controle de transações**

# Tipos da Linguagem SQL

- **DDL - *Data Definition Language* - Linguagem de Definição de Dados.**
  - São os comandos que interagem com os objetos do banco. (**CREATE, ALTER e DROP**)
- **DML - *Data Manipulation Language* - Linguagem de Manipulação de Dados.**
  - São os comandos que interagem com os dados dentro das tabelas. (**INSERT, DELETE e UPDATE**)
- **DTL - *Data Transaction Language* - Linguagem de Transação de Dados.**
  - São os comandos para controle de transação. (**BEGIN TRANSACTION, COMMIT E ROLLBACK**)
- **DCL - *Data Control Language* - Linguagem de Controle de Dados.**
  - São os comandos para controlar a parte de segurança do banco de dados. (**GRANT, REVOKE E DENY**)
- **DQL - *Data Query Language* - Linguagem de Consulta de dados.**
  - São os comandos de consulta. (**SELECT**)
    - *\*Em alguns livros o SELECT fica na DML em outros tem esse grupo próprio.*

# Visão geral



# DDL – Data Definition Language

- CREATE
- ALTER
- DROP



# DML – Data Manipulation Language

- INSERT
- UPDATE
- DELETE



# DCL – Data Control Language

- **GRANT** - dá aos usuários privilégios de acesso ao banco de dados.
- **REVOKE** - retira os privilégios de acesso do usuário dados usando o comando GRANT.



# DTL – Data Transaction Language

- **COMMIT** - compromete uma transação.
- **ROLLBACK** - reverte uma transação no caso de ocorrer algum erro.
- **SAVEPOINT** - define um ponto de salvamento dentro de uma transação.
- **SET TRANSACTION** - especifica as características da transação.





# DQL – Data Query Language

- SELECT
  - FROM
  - WHERE



# DQL - Operadores

- \*
- DISTINCT
- AS (ALIAS)
- LIMIT
- NULL
- BETWEEN
- ORDER BY
- HAVING
- CAST
- CONVERT



# DQL – Operadores+

- UNION
- INTERSECT
- EXCEPT
- DATE



# DQL – Operadores+

- LIKE
- UPPER
- LOWER
- TRIM
- REPLACE
- LPAD (REPLICATE)
- RPAD
- ...



# DQL – Operadores Agregação

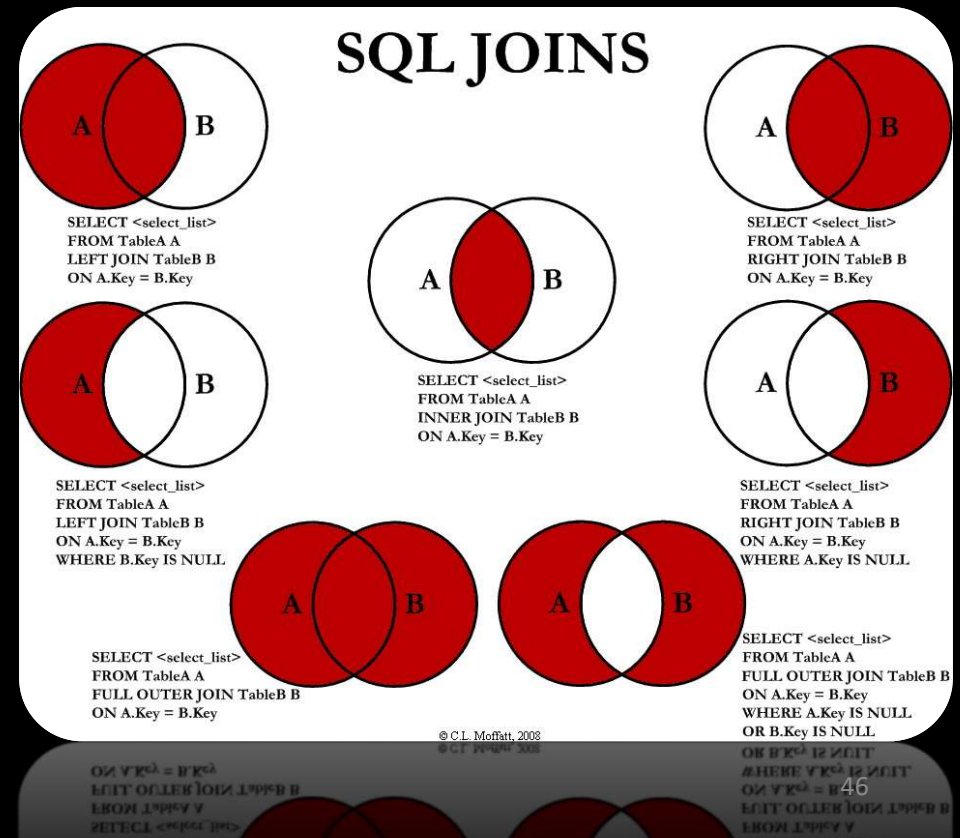
- SUM
- AVG
- MAX
- MIN
- COUNT

\*GROUP BY



# DQL - Relacionamentos (JOINS)

- INNER JOIN
- LEFT JOIN
- RIGHT JOIN
- OUTER JOIN



# Cardinalidade

[Vídeo explicando...](#)

- 1 para N
- N para 1
- N para N
- 1 para 1

São quatro as cardinalidades:

**1. 1 para N**



**2. N para 1**



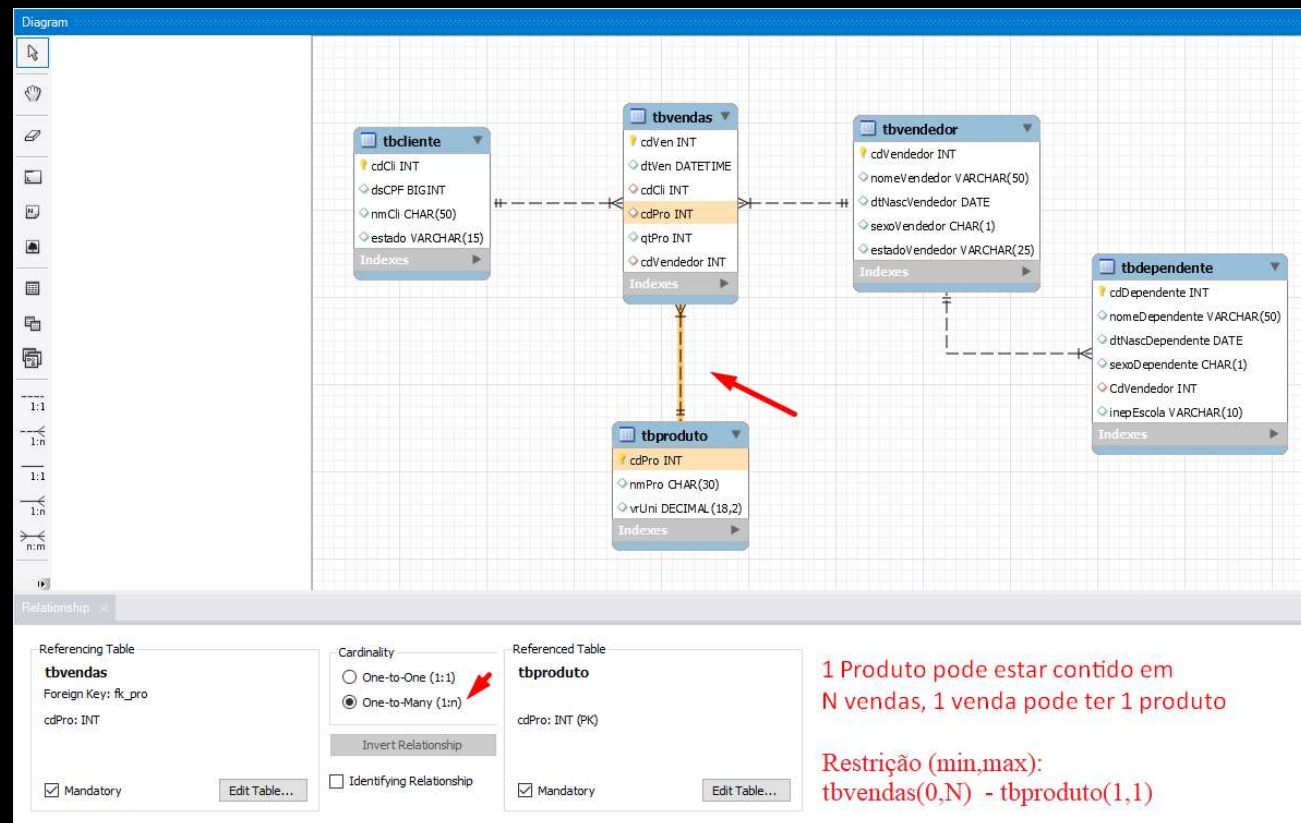
**3. N para N**



**4. 1 para 1**



# Cardinalidade (nosso exemplo)





# DQL - Controle de Fluxo

- IF
- CASE



## DQL - Subconsultas

- `SELECT * FROM tbABC  
WHERE ID IN (SELECT ID FROM tbABD);`

# DQL - Views

- CREATE VIEW vwABC AS  
SELECT ID, NOME FROM tbABC;



# AVANÇANDO...

- Enriquecimento de dados
- Relacionar tabelas em *databases* distintos

# AVANÇANDO...

- Window Functions



**Projeto**

