


Sistemas Gerenciadores de

BANCO DE DADOS



BANCOS DE DADOS SÃO MAIS IMPORTANTES NAS NOSSAS VIDAS DO QUE A GENTE IMAGINA

O que aconteceria se não houvessem sistemas
computacionais com bancos de dados?



QUE DADOS PESSOAIS ESTÃO SALVOS EM BANCOS DE DADOS?

NOME

PLACA DO
CARRO

FOTO
PUBLICADA

ENDEREÇO

HISTÓRICO
DE
COMPRAS
DO
CARTÃO

ESTADIA
EM UM
HOTEL

NÃO NOS DAMOS CONTA DE SUA PRESENÇA



INVISÍVEL

Do ponto de vista do usuário, o banco de dados deve ser invisível, e não se distinguir da aplicação que o acessa



ONIPRESENTE

Caixa eletrônico, compra de passagem aérea, sistema de e-mail, compras online, redes sociais, aplicativos, etc.

01

BANCO DE DADOS

O que é

BANCO DE DADOS

- **É UM CONJUNTO DE DADOS**
 - Planilha com dados do cliente
 - Caderno do dono do mercadinho
- Especificações de um BD
 - Reúne informações de forma organizada
 - Informações são consistentes e protegidas
 - Acesso a elas é feito em tempo hábil

BANCO DE DADOS

- **É uma coleção de dados inter-relacionados, representando informações sobre um domínio específico**
- Sala de arquivos eletrônica
- Informações que podem ser necessárias futuramente
- O que podemos fazer com essas informações?
 - Ler os dados
 - Inserir dados
 - Manipular os dados
- É uma das partes mais importantes de um sistema
- **ONDE USAMOS BANCO DE DADOS?**

VANTAGENS



Melhora no relacionamento e
na produtividade da empresa



Redução dos riscos de operação
(alinhamento da equipe)



Aumento da segurança



Melhora na tomada de
decisão



COMO ARMAZENAMOS OS DADOS SE NÃO TEMOS BANCO DE DADOS?

02

SGBD

Sistema Gerenciador de Banco de Dados

SGBD



O QUE É

Sistema Gerenciador de Banco de Dados



EXEMPLOS

PostgreSQL, MySQL, SQLServer, Oracle, entre outros



O QUE ELE FAZ

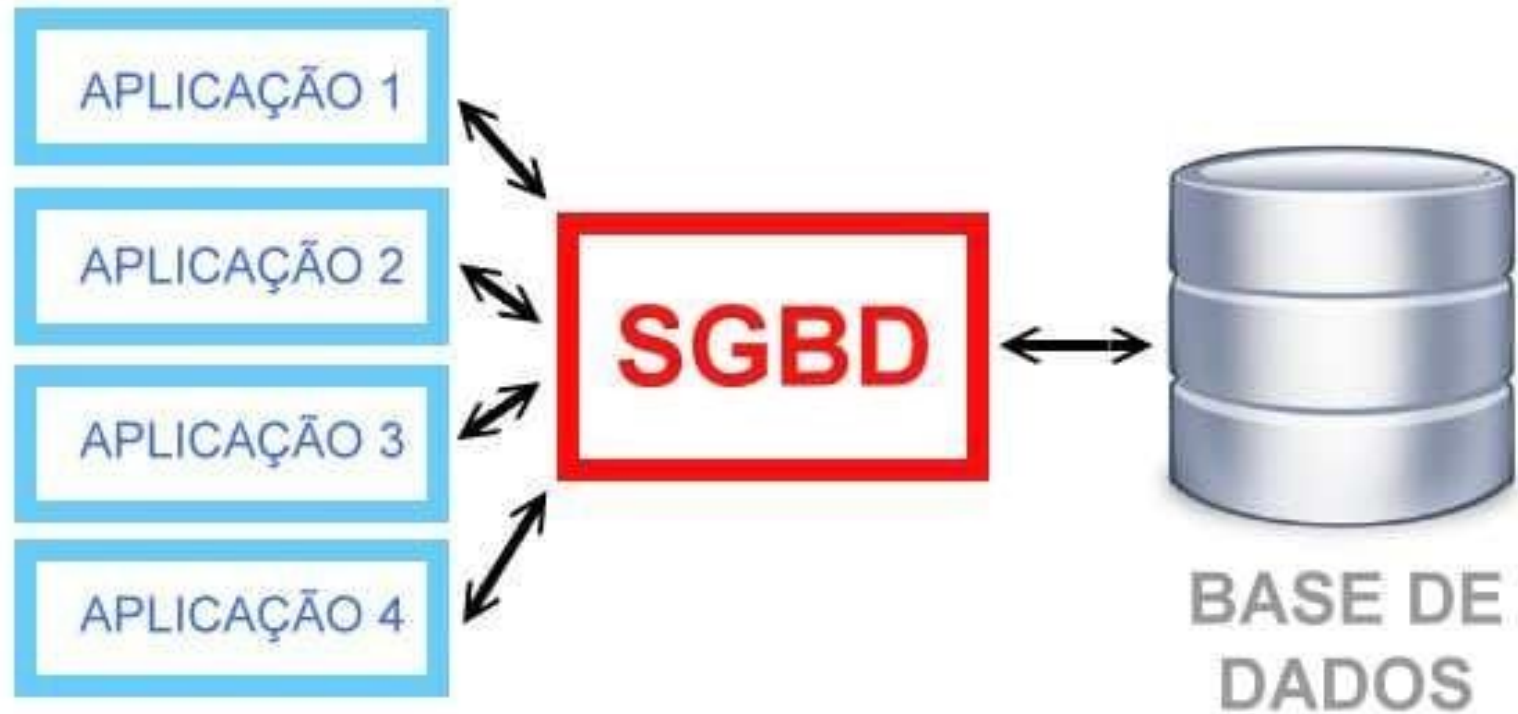
Armazena, organiza, protege, atualiza, acrescenta, exclui e acessa dados



OBJETIVOS

Isolar os usuários dos detalhes internos do banco de dados (abstrair as informações)

Promover independência dos dados com relação à aplicação



SGBD

SEGURANÇA

Regras que definem quem pode acessar a base de dados e que operações eles podem realizar

SEGURANÇA FÍSICA

As informações não podem ser acessadas por pessoas não autorizadas

SEGURANÇA LÓGICA

Sua proteção pode ser feita por meios lógicos (senhas)

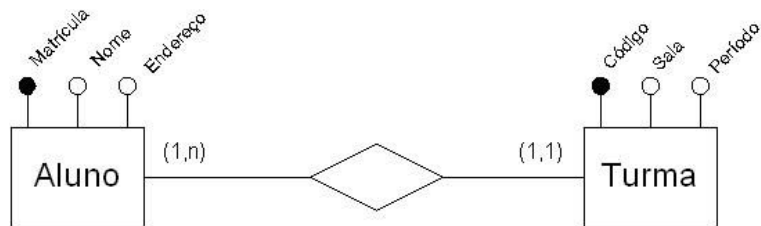
INTEGRIDADE

Garante a validade dos dados dentro do banco de dados
Exemplo: transação (transferência em uma conta bancária)

PROJETO

MODELAGEM CONCEITUAL

Define os dados que aparecerão no BD, sem se preocupar com sua implementação



MODELAGEM LÓGICA

Descreve o BD a nível lógico, dependendo do tipo de SGBD que será usado

```
1 | Aluno(mat_aluno, nome, endereco)
2 | Turma (cod_turma, sala, periodo)
```

COMO CRIAR BANCOS DE DADOS

Projetar as informações que serão manipuladas

Analisar o melhor sistema de banco de dados para o projeto

Criar uma **modelagem conceitual**

Criar o **projeto lógico**

Aplicar a **engenharia reversa**, se for otimizar bancos já existentes

03

PRINCIPAIS SGBDs DO MERCADO MUNDIAL

Nem sempre o SGBD da moda é o ideal para o seu produto

MySQL

É um dos mais utilizados

OpenSource

É muito simples, e roda em diferentes SOs

Utiliza a linguagem SQL

Utilizado pelo YouTube, Facebook, Twitter, Google, NASA



ORACLE DATABASE



Lançado no final da década de 70

É uma empresa de tecnologia que possui vários produtos na área

O banco de dados não é OpenSource

Linguagem de programação: PL/SQL

As versões trazem diferentes funcionalidades (ideias para diferentes modelos de negócio)

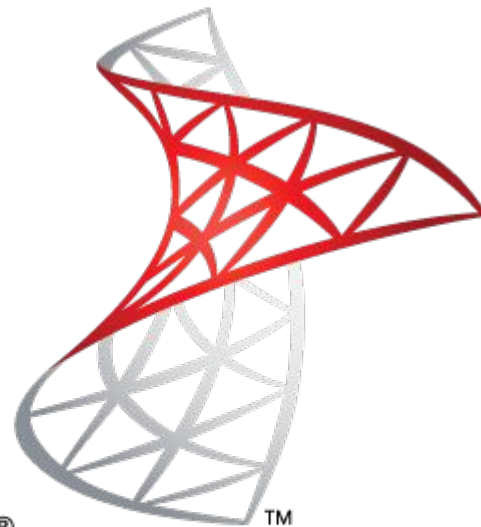
Focado em empresas de médio e grande porte

SQL Server

Lançado em 1988 como parte do Windows NT

Hoje em dia é comercializado separadamente

Diferencial: Pode-se utilizar outras linguagens de programação (C#, BASIC, .NET), e não só SQL



Microsoft®
SQL Server®

PostgreSQL



Lançado em 1989

Se mantém entre os 5 mais utilizados do mundo

OpenSource

Muito utilizado em sistemas web

Muito completo quando se trata de
funcionalidades

Utilizado pela Apple, Skype e Metrô-SP

MongoDB

Cresceu muito nos últimos anos

É um dos bancos de dados NoSQL mais utilizados no mundo

É orientado a documentos (formato JSON) - a informação é representada por um arquivo JSON

Foi criada com BigData em mente



SQLite



OpenSource

Opera sem a necessidade de um servidor

Base de dados embutida (opera como parte do aplicativo)

Suporta poucos tipos de dados (Blob, Integer, Null, Text e Real)

Facilmente transportável

Armazena as informações em um arquivo único

Mais indicada para bancos de dados menores

Indicado para desenvolvimento básico e testes

Não é adequado para o uso com múltiplos acessos (não possui as funcionalidades para gerir isso)

STACKOVERFLOW SURVEY 2025

<https://survey.stackoverflow.co/2025/>

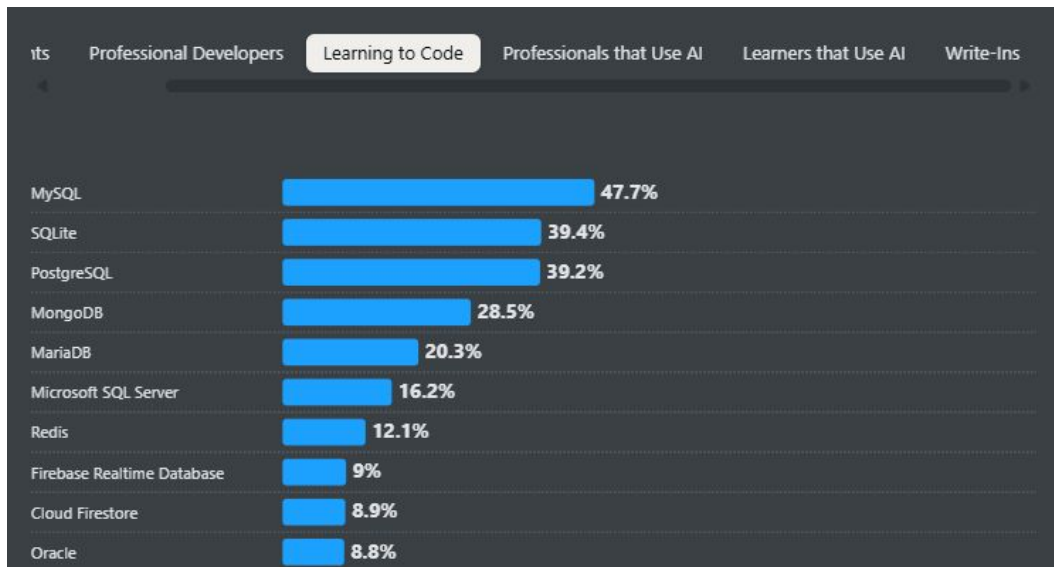
- Por que PostgreSQL é tão usado?
- MySQL vs. PostgreSQL
 - Velocidade
 - Funcionalidades
- Por que SQLite é tão usado?

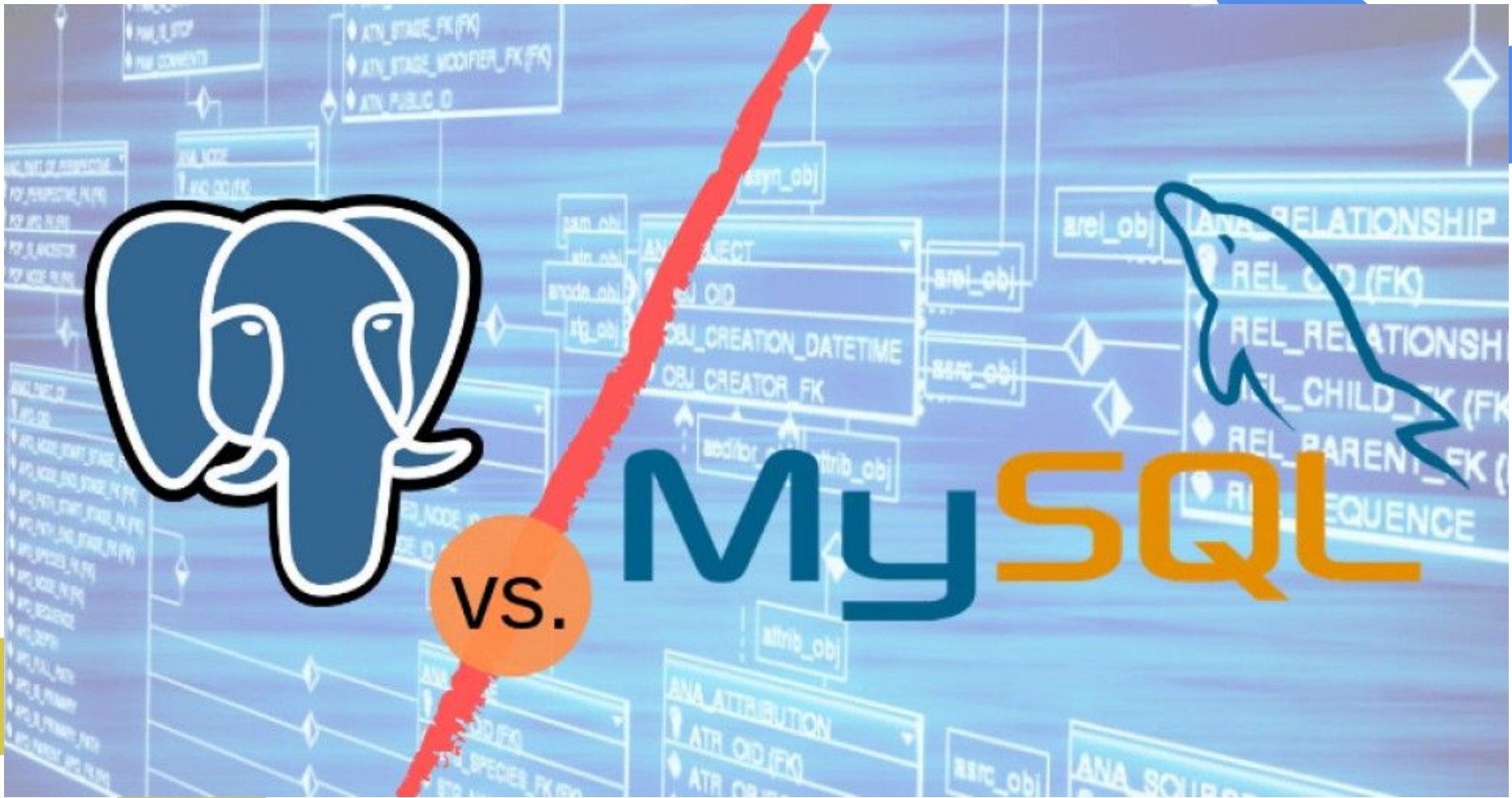


STACKOVERFLOW SURVEY 2025

<https://survey.stackoverflow.co/2025/>

- Por que PostgreSQL é tão usado?
- MySQL vs. PostgreSQL
 - Velocidade
 - Funcionalidades
- Por que SQLite é tão usado?





MySQL vs. PostgreSQL

Comparação

Velocidade

MySQL: geralmente conhecido por ser mais rápido com comandos somente leitura.

PostgreSQL: melhor com operações de leitura-escrita, conjuntos de dados massivos e consultas complicadas.

Comparação

Arquitetura

MySQL: Relacional

PostgreSQL: Objeto-relacional

PostgreSQL

Tipos de dados

Mais tipos de dados, mais operadores

MySQL

Não suporta alguns comandos SQL

LIMIT, ALL, INTERSECT, OUTER JOIN

SQLite

Por que é tão usado?

Quando usar?

SQLite

- Pequenos aplicativos independentes
- Projetos menores que não exigem escalabilidade
- Quando se têm uma necessidade de ler e escrever direto da unidade de armazenamento
- Desenvolvimento básico e testes
- Fácil de usar
- Em várias aplicações, ele já vem instalado
- **Não possui nenhum serviço de segurança embutido**

Como configurar o SGBD em uma aplicação?

Django

```
# settings.py
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'OPTIONS': {
            'read_default_file': '/path/to/my.cnf',
        },
    },
}
```

```
# my.cnf
[client]
database = NAME
user = USER
password = PASSWORD
default-character-set = utf8
```

Engine:

'django.db.backends.postgresql'

'django.db.backends.mysql'

'django.db.backends.sqlite3'

'django.db.backends.oracle'

Node.js

```
var mysql = require('mysql');

var con = mysql.createConnection({
  host: "localhost",
  user: "yourusername",
  password: "yourpassword"
});

con.connect(function(err) {
  if (err) throw err;
  console.log("Connected!");
});
```

```
const { Client } = require('pg');
client = new Client({
  host: '/cloudsql/myproject:zone:mydb',
  user: 'username',
  password: 'password',
  database: 'database_name',
});
```

Spring

```
spring.jpa.hibernate.ddl-auto=update
spring.datasource.url=jdbc:mysql://${MYSQL_HOST:localhost}:3306/db_example
spring.datasource.username=springuser
spring.datasource.password=ThePassword
```

```
spring.main.banner-mode=off
logging.level.org.springframework=ERROR
```

```
spring.jpa.hibernate.ddl-auto=none
```

```
spring.datasource.initialization-mode=always
spring.datasource.platform=postgres
spring.datasource.url=jdbc:postgresql://localhost:5432/testdb
spring.datasource.username=postgres
spring.datasource.password=s$cret
```

```
spring.jpa.properties.hibernate.jdbc.lob.non_contextual_creation=true
```


DÚVIDAS?

Qualquer dúvida entrar em
contato

Profª. Carla Fernandes Curvelo
carla.fernandes@ufrn.br