

# Introdução à Base de Dados



Prof. Jeferson Souza, MSc.

(jefecomp)

[jefecomp.official@gmail.com](mailto:jefecomp.official@gmail.com)



**UDESC**  
UNIVERSIDADE  
DO ESTADO DE  
SANTA CATARINA

JOINVILLE  
CENTRO DE CIÊNCIAS  
TECNOLÓGICAS

# Definição

O que é uma base de dados?



# Definição

O que é uma base de dados?



- ▶ Uma base de dados pode ser definida como um conjunto de arquivos organizados de forma a armazenar dados relevantes para usuários/aplicações;
- ▶ Novos dados podem ser inseridos na base de dados, assim como dados existentes podem ser atualizados, removidos, ou consultados.

# Definição

Como esses dados podem ser manipulados?



## ***Sistema de Gerenciamento de Base de Dados (SGBD)***

Através de um Sistema de Gerenciamento de Base de Dados  
(SGBD).

## Um pouco de história ...

- ▶ 1950 e 1960: Tempo das fitas magnéticas para armazenamento de dados. Operações de leitura e escrita das fitas eram realizadas sequencialmente;
- ▶ 1960 e 1970: Introdução do uso de discos rígidos, melhorando consideravelmente a tarefa de acesso e processamento de dados. Dados podiam ser acessados diretamente. Nessa época foi publicado o artigo seminal que define o modelo relacional;

## Um pouco de história ...

- ▶ 1980: Implementation do projeto R pela IBM, o qual criou um eficiente sistema de gerenciamento de base de dados relacional;
- ▶ 1990 (Início): Surge a linguagem de consulta estruturada SQL, do inglês Structured Query Language;

## Um pouco de história ...

- ▶ 1990: Crescimento da World Wide Web, e como consequência crescimento na utilização de sistemas de gerenciamento de bases de dados;
- ▶ 2000: Popularização no uso de bases de dados de código aberto (ex: PostgreSQL) e surgimento de novas tecnologias de bases de dados NoSQL.

# Definição de SGBD



## ***Sistema de Gerenciamento de Base de Dados (SGBD)***

O que é um SGBD?

Um SGBD é um conjunto de arquivos que armazenam dados interrelacionados, juntamente com um conjunto de programas que permitem manipular esses dados.



# Onde estão os SGBDs?

SGBDs são como os brasileiros: estão em todo o lado :-)!

- ▶ Grandes corporações (ex: Google, Facebook, ...);
- ▶ Diferentes segmentos: Indústria automotiva, indústria de entretenimento, bancos, entre outros.

# Onde estão os SGBDs?

SGBDs são como os brasileiros: estão em todo o lado :-)!

- ▶ Grandes corporações (ex: Google, Facebook, ...);
- ▶ Diferentes segmentos: Indústria automotiva, indústria de entretenimento, bancos, entre outros.

Lembrem-se:

Armazenamento de dados permite extração de informação.

# Onde estão os SGBDs?

SGBDs são como os brasileiros: estão em todo o lado :-)!

- ▶ Grandes corporações (ex: Google, Facebook, ...);
- ▶ Diferentes segmentos: Indústria automotiva, indústria de entretenimento, bancos, entre outros.

Lembrem-se:

Armazenamento de dados permite extração de informação.

Informação é PODER!!!

# Benefícios dos SGBDs

- ▶ Reduzir redundância e inconsistência nos dados armazenados;
- ▶ Melhoria no acesso aos dados - menos complicado e mais eficiente;
- ▶ Interface única para acesso a diferentes tipos de dados;
- ▶ Manutenção da integridade dos dados;
- ▶ Suporte ao acesso simultâneo dos dados;
- ▶ Suporte a segurança dos dados.

# Benefícios dos SGBDs

- ▶ Reduzir redundância e inconsistência nos dados armazenados;
- ▶ Melhoria no acesso aos dados - menos complicado e mais eficiente;
- ▶ Interface única para acesso a diferentes tipos de dados;
- ▶ Manutenção da integridade dos dados;
- ▶ Suporte ao acesso simultâneo dos dados;
- ▶ Suporte a segurança dos dados.

O que eu ganho utilizando um SGBD?

Uma visão abstrata no acesso aos dados, sem a necessidade de saber os detalhes técnicos do armazenamento.



# Nível Físico (Physical Level)

## Nível Físico (Physical Level)

Detalhes de armazenamento dos dados são especificados no nível físico. O nível físico define as estruturas de dados necessárias para armazenamento, e organização dos dados no dispositivo de armazenamento (ex: disco rígido);

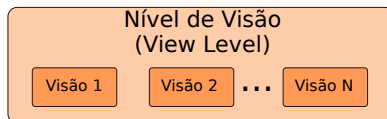
# Nível Lógico (Logical Level)

## Nível Lógico (Logical Level)

Primeiro nível de abstração dos dados armazenados no nível físico. No nível lógico são especificados os relacionamentos e restrições na definição dos dados. Nesse nível é definido o que realmente será armazenado.



# Nível de Visão (View Level)



Define diferentes visões da mesma base de dados. Além disso pode especificar restrições de acesso, ou seja, um determinado usuário pode ter permissão de acesso somente a uma pequena parte da base de dados.

# O que é um Modelo de Dados?

Um modelo de dados é uma forma de descrever dados ou informação nos três níveis de abstração: físico, lógico, e visão.

# O que é um Modelo de Dados?

Possui 3 principais características:

- ▶ Estrutura dos dados;
- ▶ Operações sobre os dados;
- ▶ Restrições sobre os dados.

# Modelos de dados

- ▶ Relacional;
- ▶ Entidade Relacionamento (E-R);
- ▶ Baseado em objetos;
- ▶ Objeto-Relacional;
- ▶ Semi-estruturado.

Existem diferentes modelos de dados na literatura. Vamos nos concentrar (por agora) no modelo de dados Relacional.

# O que é o modelo de dados relacional?

- ▶ Consiste em uma forma estruturada de representar dados, e as relações entre esses dados;
- ▶ Dados são representados em tabelas, e uma tabela define um domínio específico desses dados;
- ▶ As colunas das tabelas representam as suas características (e tipos de dados).

# Tabelas

Usuário		
ID	Nome	Email
1	Jeferson	jef@email.com

Registro

Um registro é uma linha em uma tabela que possui valores atribuídos a cada uma das colunas especificadas. Esses valores podem ser nulos (ausência de valor).

# Identificando registros unicamente

Por que é importante identificar registros em uma tabela de forma única?

- ▶ Evitar inserção de dados repetidos (diminuir redundância);
- ▶ Permitir o estabelecimento de relações com outras tabelas;
- ▶ Permitir uma consulta mais eficiente dos dados.

A identificação única de um registro em uma tabela é denominada chave primária.

# Chave primária (primary key)

- ▶ Identifica unicamente um registro em uma tabela;
- ▶ Pode ser composta por uma ou mais colunas de uma tabela. Ex: id, {nome, telefone};
- ▶ Uma boa chave primária praticamente não se altera com o passar do tempo. Exemplo: identificadores sequenciais gerados automaticamente.



# Chave primária (primary key)

Exemplo:

Chave primária



Usuário		
ID	Nome	Email
1	Jeferson	jef@email.com

# Candidatos à chave

O que são candidatos à chave (candidate keys)?

Candidatos a chave são colunas (ou combinação de colunas) de uma tabela que podem (além da chave primária) identificar unicamente um registro da tabela. Ex: CPF, email, {nome, telefone}.

# Candidatos à chave

Exemplo:



# Candidatos à chave

Por que é importante identificar colunas candidatas à chave?

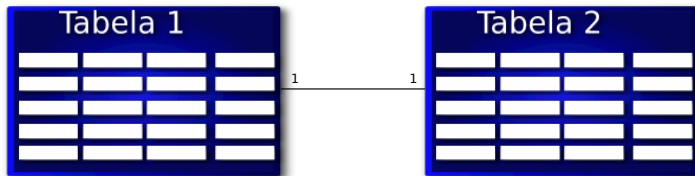
Colunas candidatas à chave são ótimos elementos para gerar índices (tópico abordado mais a frente no curso).

# Chave estrangeira (foreign key)

- ▶ Referencia a chave primária de outra tabela;
- ▶ Usar chaves estrangeiras é a forma de estabelecer relações entre tabelas no modelo relacional.

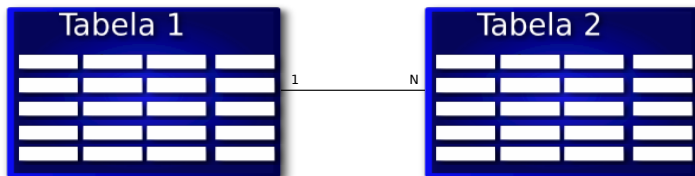


# Relacionamento Um para Um (One to One)



A chave estrangeira é definida do lado da relação que fizer mais sentido para consulta dos dados.

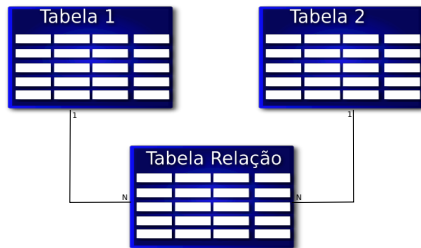
# Relacionamento Um para Muitos (One to Many)



A chave estrangeira é definida do lado N da relação.



# Relacionamento Muitos para Muitos (Many to Many)



A chave estrangeira de ambas as tabelas é definida na tabela de relação. Asumindo que PK1 é a chave primária da tabela 1, e PK2 a chave primária da tabela 2, podemos definir a chave primária da tabela de relação como sendo:

$$\{ PK1, PK2 \}$$

# Schema de base de dados

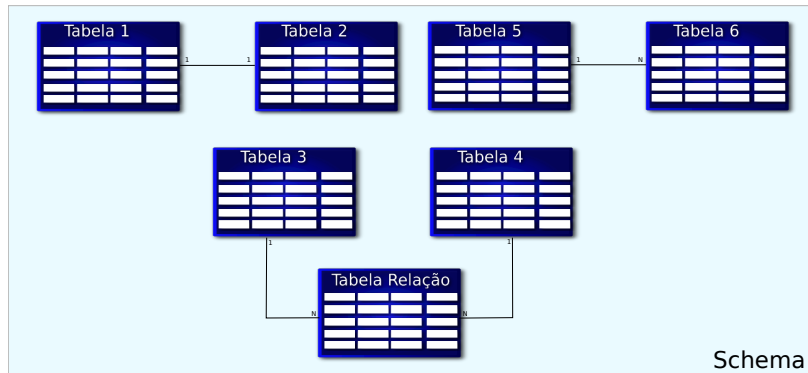
- ▶ Define todos os elementos presentes em uma base de dados (Estutura);
- ▶ Especificado com auxílio da linguagem de definição de dados, do inglês Data Definition Language (DDL) (sublinguagem do SQL);
- ▶ Depois de completamente definido, mudanças em um dado schema praticamente não ocorrem.
- ▶ Três tipos de schema: Físico, Lógico, e Visão. Seguem a mesma organização dos níveis de abstração apresentados a partir do slide 9.

# Schema de base de dados (Continuação)

## Instância de um schema

Uma instância de um schema representa toda a coleção de dados desse schema em um dado instante de tempo.

# Exemplo de schema de base de dados





# Exercício de Fixação (Continuação)

Depois de realizada a modelagem, responda (Sim ou Não) a seguinte pergunta:

A estratégia utilizada para converter o diagrama E-R para tabelas no banco de dados foi representar cada entidade como uma uma tabela?

# Fases de Projeto

- ▶ Levantamento inicial de requisitos;
- ▶ Desenho conceitual;
- ▶ Especificação dos requisitos funcionais;
- ▶ Desenho lógico;
- ▶ Desenho físico.

## Fase de levantamento inicial de requisitos

- ▶ Iteração com especialistas no domínio dos dados que o projeto da base de dados está inserido;
- ▶ Iteração com usuários da base de dados;
- ▶ Identificação e levantamento de requisitos proveniente da interação com ambos especialistas e usuários;
- ▶ Produção da especificação dos requisitos de usuário;
- ▶ Remoção de redundâncias existentes nos requisitos especificados.



# Fase de desenho conceitual

- ▶ Criação do schema conceitual da base de dados, o qual especifica a estrutura dos dados e suas relações;
- ▶ Validação do schema conceitual criado para confirmar que todos os requisitos levantados estão presentes no modelo conceitual, e não tem conflitos;
- ▶ Remoção de eventuais redundâncias de dados presentes no schema conceitual criado.

# Fase de Especificação de Requisitos Funcionais

- ▶ Descrição das operações e transações que poderão ser executadas sobre os dados;
- ▶ Operações podem incluir inserção, atualização, remoção e busca de dados;
- ▶ Verificação do schema conceitual visando atestar que, o mesmo, encontra-se em conformidade com os requisitos funcionais descritos.

# Fase de Desenho Lógico

- ▶ Nesta fase, acontece o mapeamento do modelo conceitual mais abstrato (alto-nível) para um modelo concreto de dados;
- ▶ O resultado é a implementação do schema conceitual na base de dados.

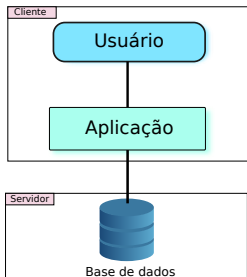
# Fase de Desenho Físico

- Descrição de estruturas de dados (à nível físico) que permitam o armazenamento e acesso eficiente da implementação do schema conceitual realizada no nível lógico.

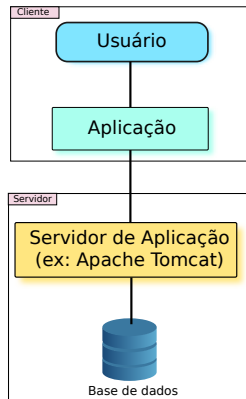
## Observação

Normalmente não é necessário especificar estruturas de dados para auxiliar no armazenamento do schema conceitual implementado ao nível lógico. Porém, em alguns casos, pode ser necessário adicionar novas estruturas de dados no nível físico para permitir tanto o armazenamento, quanto o acesso eficiente dos dados no nível físico.

# Tipos de arquitetura de banco de dados



## Arquitetura de duas camadas



## Arquitetura de três camadas

\*Baseada na figura 1.6 do livro “Database Systems” [SilberchatzEtAl, 2011].

# Bibliografia

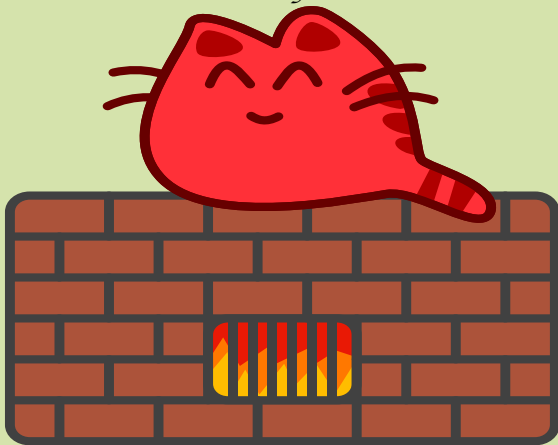


Garcia-Molina, H. and Ullman, J. D. and Widom, J.  
“Database Systems: The Complete Book. 2nd edition.  
Prentice Hall, 2008.



Silberschatz, A. and Korth, H.F. and Sudarshan, S.  
“Database Systems”. 6th edition. McGrawHill, 2011.

*That's it folks!*



*Thank you for your attention!*