

INSTITUTO FEDERAL  
PIAUÍ  
Campus Parnaíba

# Introdução a Banco de Dados

Prof. Msc Denival A. dos Santos

# 30

- **Dados** são tudo que podemos inferir ou coletar sobre uma situação específica.
- Sozinho, dissociado de um contexto, não expressa algo que traga qualquer certeza ou elimine dúvidas de qualquer natureza. No máximo, pode-se deduzir que seja um referencial.

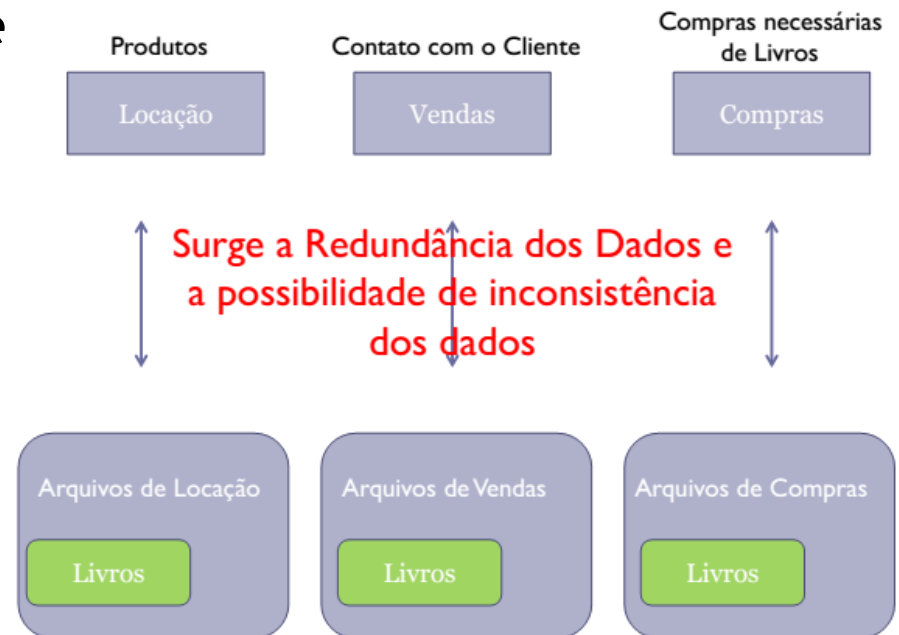
# Informação

- Os dados reunidos passam a apresentar um significado, de tal maneira que podem ser interpretados pelas pessoas, produzindo **informação**.
- Exemplo:
  - Lista de clientes com seus respectivos nomes, números de CPF, endereços ordenados pelo nome do cliente.

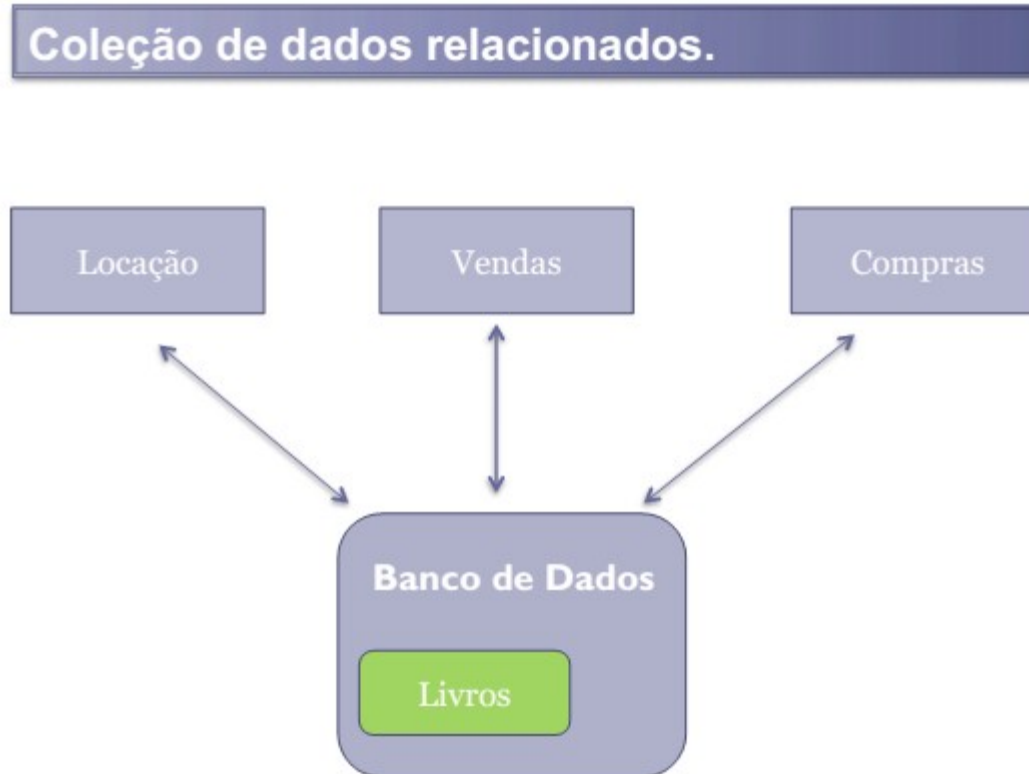
**Observação:** Para se gerar informação, no mínimo, é necessário conhecer (**Conhecimento**) antecipadamente o contexto essencial para o qual ela servirá.

# Banco de Dados - Redundância - Problema

- Limitações na utilização de dados em arquivos
  - Dificuldade de acesso
    - Problemas para filtrar ou agregar dados
  - Redundância e inconsistência
  - Isolamento dos dados
  - Anomalias de acesso concorrente
  - Problemas de segurança.

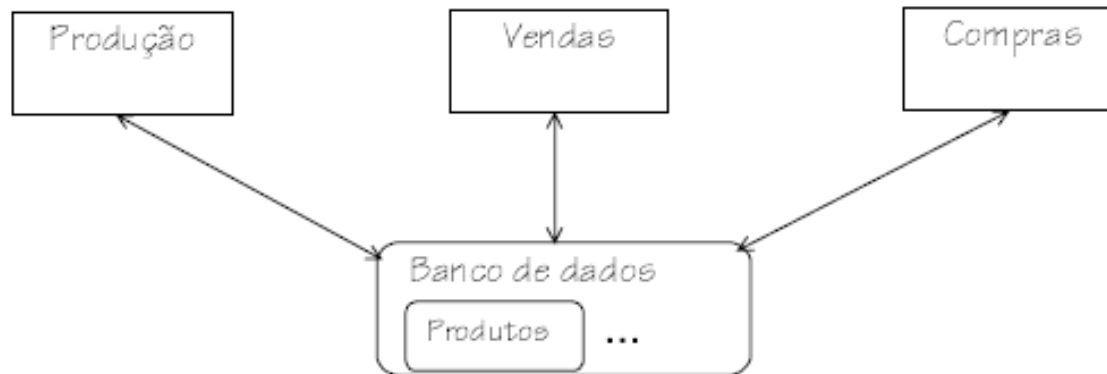


# Banco de Dados - Solução



# Banco de Dados - Definição

- **Banco de dados** é uma coleção de dados referentes a um assunto ou propósito específico, com o objetivo de organizar os dados de modo a tornar a vida dos usuários do negócio em questão mais prática, precisa, rápida e confiável.



# Banco de Dados

- Que banco de dados temos ao nosso redor?
  - Agenda de contatos do celular.
  - Firefox - Favoritos, histórico.
    - Armazenados no SQLite.
  - Caixas eletrônicos.
  - Mecanismos de busca.
  - Postagens nas redes sociais.

# SGBD - Sistema Gerenciador de Banco de Dados

- **SGBD** - É uma coleção de programas que permite aos usuários criar e manter um banco de dados.
- O principal objetivo de um SGBD é proporcionar um ambiente tanto conveniente quanto eficiente para a recuperação e armazenamento das informações do banco de dados.





# SGBD - Funcionamento



1. O usuário emite uma solicitação de acesso.
2. O SGBD intercepta a solicitação e a analisa.
3. O SGBD inspeciona os esquemas externos (ou sub esquemas) relacionados àquele usuário, os mapeamentos entre os três níveis, e a definição da estrutura de armazenamento.
4. O SGBD realiza as operações solicitadas no banco de dados armazenado.

# Exemplos de SGBD's

- **SQL Server** (da Microsoft)
- **PostgreSQL** (código aberto)
- **Firebird** (código aberto)
- **MySQL** (código aberto, atualmente Oracle)
- **Oracle Database** (da Oracle)
- **DB2** (da IBM)
- **MariaDB** (código aberto)



# SGBD - Popularidade

Rank			DBMS	Database Model	Score		
Mar 2019	Feb 2019	Mar 2018			Mar 2019	Feb 2019	Mar 2018
1.	1.	1.	Oracle	Relational, Multi-model	1279.14	+15.12	-10.47
2.	2.	2.	MySQL	Relational, Multi-model	1198.25	+30.96	-30.62
3.	3.	3.	Microsoft SQL Server	Relational, Multi-model	1047.85	+7.79	-56.94
4.	4.	4.	PostgreSQL	Relational, Multi-model	469.81	-3.75	+70.46
5.	5.	5.	MongoDB	Document	401.34	+6.24	+60.82
6.	6.	6.	IBM Db2	Relational, Multi-model	177.20	-2.23	-9.47
7.	9.	7.	Microsoft Access	Relational	146.20	+2.18	+14.26
8.	7.	8.	Redis	Key-value, Multi-model	146.12	-3.32	+14.90
9.	8.	9.	Elasticsearch	Search engine, Multi-model	142.79	-2.46	+14.25
10.	10.	11.	SQLite	Relational	124.87	-1.29	+10.06
11.	11.	10.	Cassandra	Wide column	122.80	-0.58	-0.69
12.	12.	15.	MariaDB	Relational, Multi-model	84.31	+0.89	+21.21
13.	13.	13.	Splunk	Search engine	83.10	+0.29	+17.44
14.	14.	12.	Teradata	Relational	75.22	-0.75	+2.76
15.	15.	18.	Hive	Relational	73.00	+0.71	+16.00
16.	16.	14.	Solr	Search engine	60.01	-0.95	-4.80

# SGBD - Funcionalidades

- Funcionalidades
  - Controle de redundância
  - Acesso limitado aos dados (segurança)
  - Armazenamento persistente dos dados
  - Definição e manutenção de restrições de integridade
  - Controle de concorrência
  - Recuperação de falhas
- Um dos maiores benefícios de um banco de dados é proporcionar ao usuário uma visão abstrata dos dados.
- Os usuários não precisam ter conhecimento sobre detalhes de implementação e manutenção dos dados.

# SGBD - Usuários

- **Usuários finais**

- São aqueles que vão trabalhar diariamente com as aplicações desenvolvidas.

- **Analistas de sistemas**

- Determinam os requisitos dos usuários finais.

- **Projetista de Banco de Dados**

- Identificam os dados a serem armazenados.
- Escolhem estruturas adequadas para representar e armazenar esses dados.

- **Programadores de aplicação**

- Implementam essas especificações como programas.

- **Administrador do Banco de Dados (DBA)**

- Gerencia os dados armazenados no SGBD.

# Instâncias e Esquemas

- Um banco de dados muda ao longo do tempo por meio de informações que nele são inseridas ou excluídas.
- Segundo Silberschatz (2006), os *esquemas* de dados dizem respeito ao projeto geral do banco de dados e é um aspecto que raramente é modificado.
- Uma *instância* do banco de dados diz respeito à coleção de dados armazenados em um banco de dados em um determinado momento (SILBERSCHATZ, 2006). A instância modifica toda vez que uma alteração no banco de dados é feita. O SGBD é responsável por garantir que toda instância do banco de dados satisfaça o seu esquema do banco de dados, respeitando sua estrutura e suas restrições.

# Modelo de Dados

- Um *modelo de (banco de) dados* é uma descrição dos tipos de informações que estão armazenadas em um banco de dados.
  - Por exemplo:
    - No caso de uma indústria, o modelo de dados poderia informar que o banco de dados armazena informações sobre produtos e que, para cada produto, são armazenados seu código, preço e descrição.
- O *projeto de banco de dados* tem o objetivo de transformar as necessidades de informações no negócio em um banco de dados.

# Modelo de Dados

- Modelo de Rede
- Modelo Hierárquico
- Modelo Entidade-Relacionamento
- Modelo Relacional
- Modelo de dados baseados em objeto
- Modelo de dados semiestruturado



# Níveis de Abstração



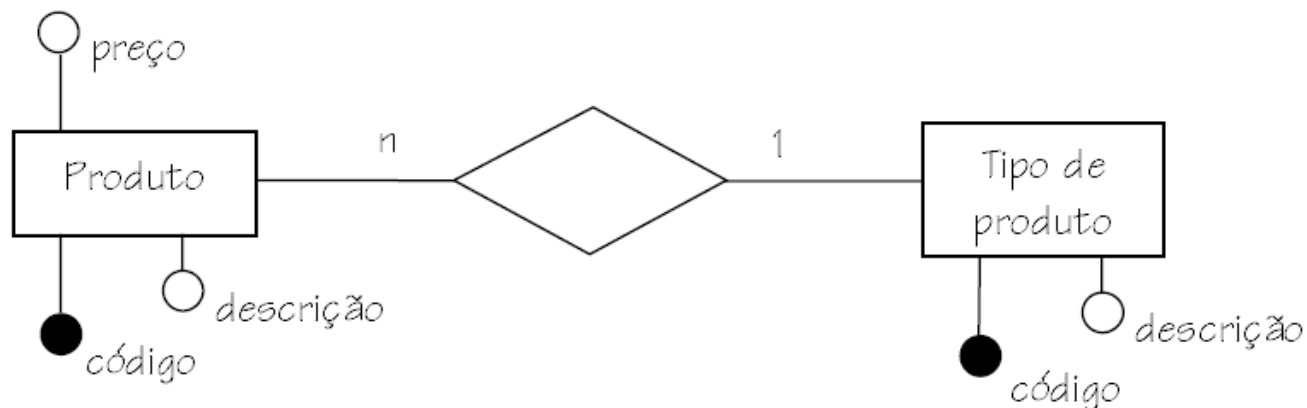
modelo **conceitual**

modelo **lógico**

modelo **físico**

# Modelo conceitual

- Um *modelo conceitual* é uma descrição do banco de dados de forma independente de implementação em um SGBD. O modelo conceitual registra que dados podem aparecer no banco de dados, mas não registra como estes dados estão armazenados a nível de SGBD.
- A técnica mais difundida de modelagem conceitual é a *abordagem entidade-relacionamento* (ER). Nesta técnica, um modelo conceitual é usualmente representado através de um diagrama, chamado *diagrama entidade-relacionamento (DER)*.



# Modelo lógico

- Um *modelo lógico* é uma descrição de um banco de dados no nível de abstração visto pelo usuário do SGBD.
- O modelo lógico é dependente do tipo particular de SGBD que está sendo usado.
- Exemplo de modelo lógico textual:

```
TipoDeProduto(CodTipoProd, DescrTipoProd)
```

```
Produto(CodProd, DescrProd, PreçoProd, CodTipoProd)  
CodTipoProd referencia TipoDeProduto
```

# Modelo Físico

- Contém detalhes de armazenamento interno de informações.
- Detalhes que:
  - Não têm influência sobre a programação de aplicações no SGBD, mas, influenciam a performance da aplicações.
  - São usados por profissionais que fazem sintonia (ajuste de desempenho - “tuning”) de banco de dados.

# Linguagens de Acesso a um BD

- Podemos dividir em duas categorias os tipos de linguagem oferecidas por um SGBD. Uma delas para expressar consultas e atualizações e outra específica para os esquemas do banco de dados. São elas:
  - Linguagem de definição de dados (DDL)
    - Utilizada para o esquema de um BD.
  - Linguagem de manipulação de dados (DML)
    - Utilizada para permitir:
      - consultas sobre um BD;
      - Inserção em tabelas;
      - Remoção em uma tabela e;
      - atualização de dados em uma tabela.

# Linguagens de Acesso a um BD

- Exemplo DDL (SQL)

- create table Empregado  
(matr integer not null,  
nome varchar(35),  
salário real,  
primary key(matr))

# Linguagens de Acesso a um BD

## ▪ Exemplos DML (SQL)

- Consultas sobre um BD

- Exemplo (SQL)

```
select nome  
from Empregado  
where salario > 7000
```

- Inserções em uma tabela

- Exemplo (SQL)

```
insert into Empregado values(123, "Bárbara", 5000.00)
```

- Remoções em uma tabela

- Exemplo (SQL)

```
delete from Empregado  
where matr=14
```

- Atualizar valores de atributos de uma tabela

- Exemplo (SQL)

```
update Empregado set salário=salário*1.15  
where salário<1500.00
```