



Curso Técnico em Desenvolvimento de Sistemas Online

Banco de Dados I

GEEaD - Grupo de Estudos de Educação a Distância

Centro de Educação Tecnológica Paula Souza

Expediente

PROGRAMA NOVOTEC VIRTUAL
GOVERNO DO ESTADO DE SÃO PAULO
EIXO TECNOLÓGICO DE INFORMAÇÃO E COMUNICAÇÃO
CURSO TÉCNICO EM DESENVOLVIMENTO DE SISTEMAS
BANCO DE DADOS

Autor:

Eliana Cristina Nogueira Barion
Rute Akie Utida

Revisão Técnica:

Lilian Aparecida Bertini

Revisão Gramatical:

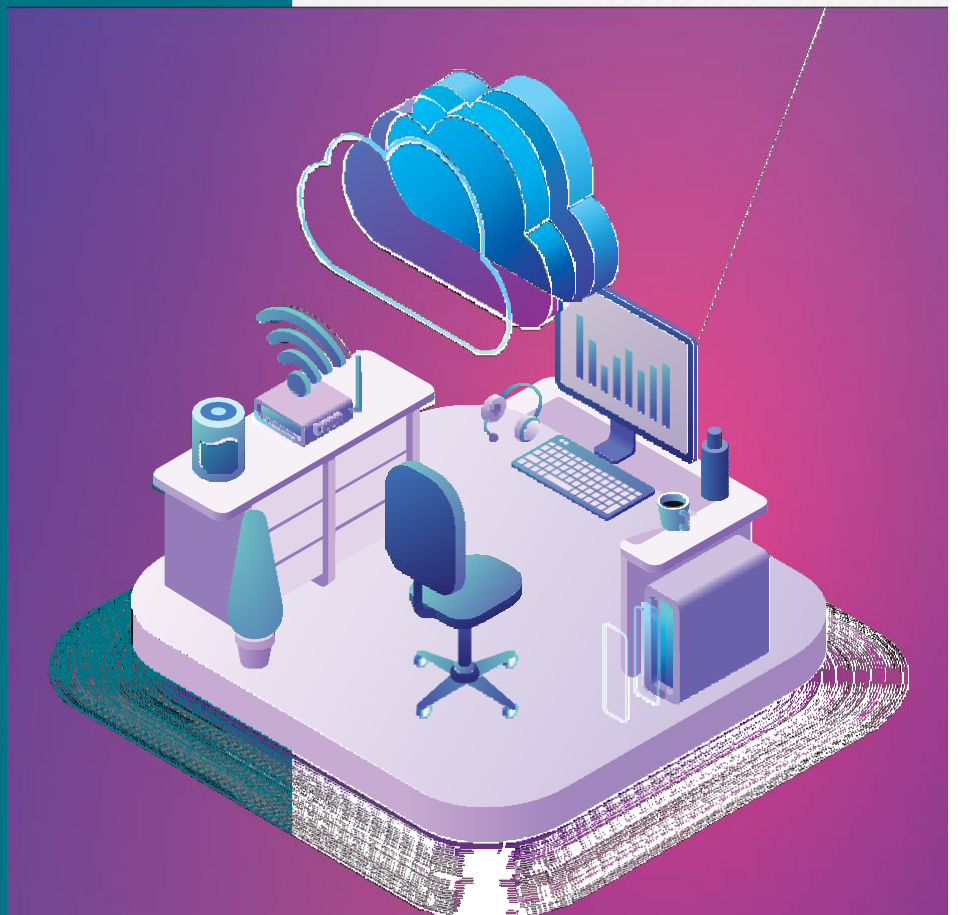
Juçara Maria Montenegro Simonsen Santos

Editoração e Diagramação: Flávio Biazim

São Paulo – SP, 2019

AGENDA 8

INTRODUÇÃO BANCO DE DADOS





MERGULHANDO NO TEMA...

Banco de Dados

O conceito de banco de dados relaciona-se ao armazenamento, manutenção e resgate de informações em um sistema. A necessidade de organizar informações está presente em diversos contextos cotidianos:

	Em casa, quando precisamos organizar documentos como manuais de eletrodomésticos ou folhetos de delivery.
	No uso de ferramentas digitais, como celulares e computadores, quando precisamos organizar arquivos virtuais.
	Em empresas, ao permitir o resgate de informações para que os gestores decidam de modo estratégico.

Em outras palavras, bancos de dados são conjuntos de informações inter-relacionadas sobre determinado assunto, armazenados de modo a permitir o acesso organizado por parte dos usuários. Segundo a definição de Elmasri e Navathe (2011)¹, “um banco de dados representa algum aspecto do mundo real, às vezes chamado de **minimundo** ou de **universo de discurso** (UoD – Universe of Discourse).”

Os bancos de dados têm como objetivo proporcionar **agilidade, uniformidade e segurança** em todas as suas fases. Ao juntarmos o conhecimento pré-existente à velocidade de processamento e à capacidade de armazenamento de informações proporcionadas pela informática, é possível criar modelos interessantes

¹ ELMASRI, Ramez; navathe, Shamkant B. **Sistemas de Bancos de Dados**. 6ª ed., edição em português. Pearson, 2011.

para o usuário: fáceis de usar, com rápida consulta e resposta, e baixo custo de manutenção².

O desenvolvedor de sistemas tem a importante tarefa de criar bancos de dados íntegros, concisos e seguros para suas aplicações.

Banco de Dados ou Banco de Informações?

Agora que você já conhece a definição e viu exemplos de banco de dados, procure responder: por que chamamos “banco de dados”? Não poderia ser “banco de **informações**”?

Dados são fatos que podem ser registrados e têm um significado.

Os **dados** constituem a matéria prima da informação, ou seja, é a informação não tratada. Por exemplo, em uma pesquisa eleitoral, cada participante fornece suas opiniões, conforme as suas preferências entre os candidatos, mas essa coleta não significa muita coisa no âmbito da eleição. Só depois de ser integrada às demais opiniões é que teremos algo com significado.

Veja outro exemplo:

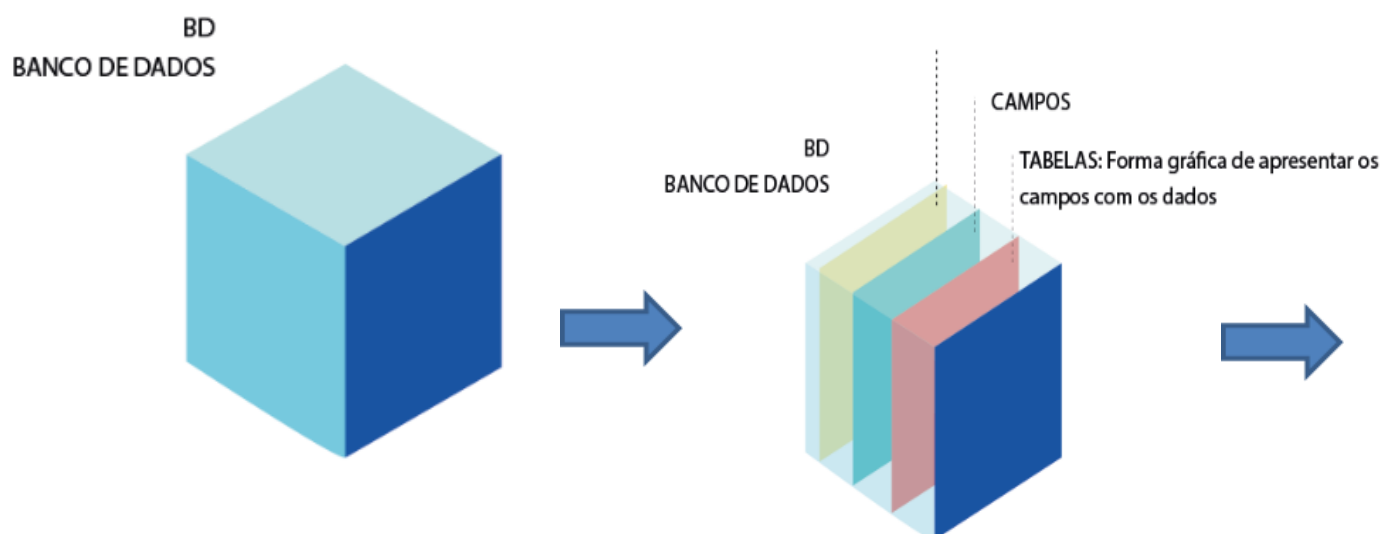
(11) 3289-3587 é um dado do número de um telefone.

Já a frase: O número do telefone da Viviane é (11) 3289-3587. Nessa frase, temos uma informação!

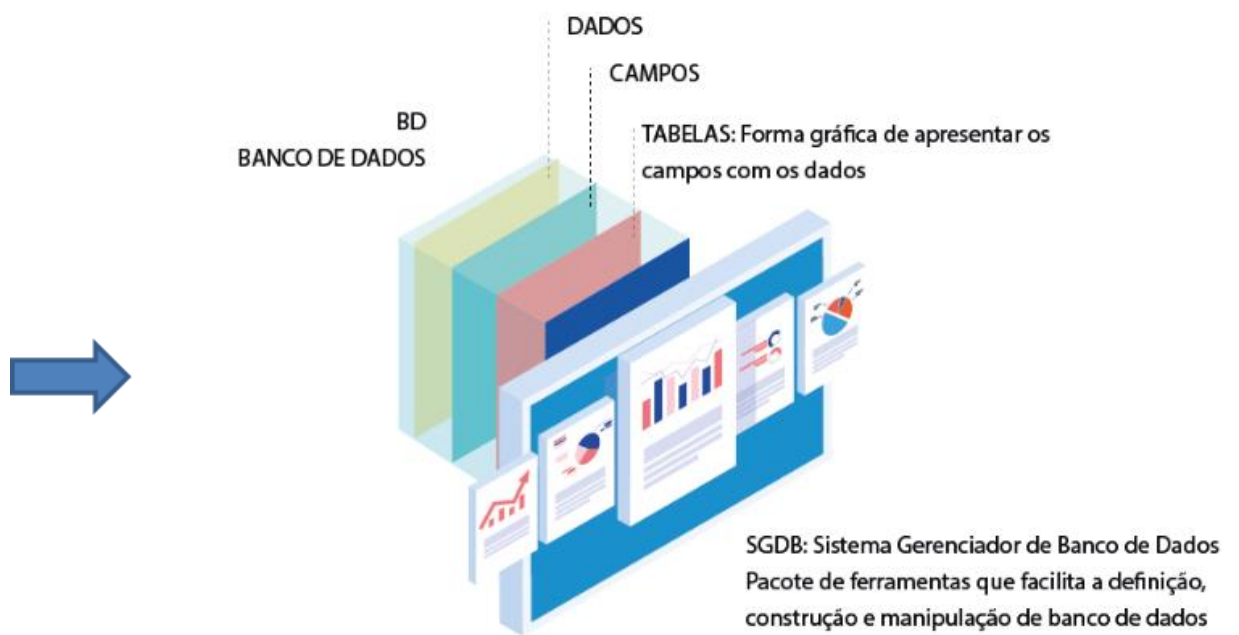
Informação é um conjunto de dados organizados, que passam algum conhecimento e referência sobre um determinado acontecimento, fato ou fenômeno

Agora vamos continuar para entender a Estrutura de um Banco de Dados, ou seja, como ele funciona.

Analise a imagem a seguir:

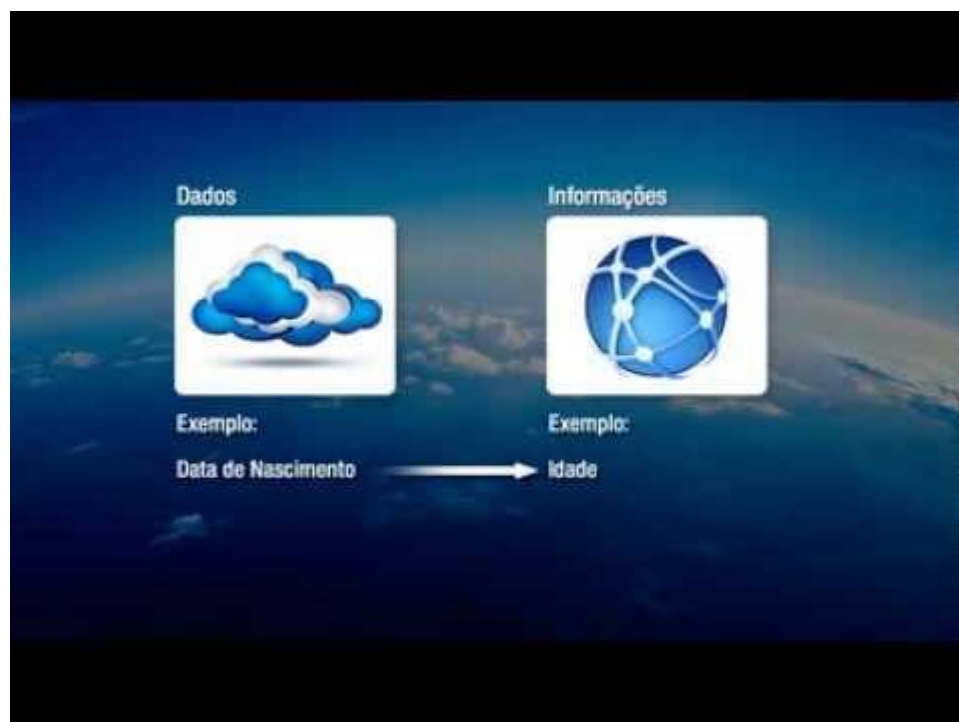


² Adaptado de Alessandro Nicoli de Mattos. Informação é prata. Compreensão é ouro, 2010



Você notou que um banco de dados é composto por dados **contidos em campos** que são **organizados em tabelas**? Também percebeu a importância do Sistema Gerenciador do Banco de Dados? Ele é um programa que permite ao usuário interagir com o banco, ou seja, permite que o usuário inclua, manipule e consulte sua base de dados.

Assista ao vídeo que retoma alguns conceitos de Banco de Dados. Disponível em <https://www.youtube.com/watch?v=hap3zFj3zSg>



Compreender a estrutura dos bancos de dados ajudará você a entender como funcionará a instalação do software SGBD e a linguagem SQL na prática. Vamos abordar esses conceitos para que você possa entender melhor!

Sistema Gerenciador de Banco de Dados (SGBD)

Como você viu, um Sistema Gerenciador de Banco de Dados (SGBD) é um pacote de ferramentas com o propósito geral de **facilitar os processos de definição, construção, compartilhamento e manipulação de um banco de dados entre vários usuários e aplicações.**

A construção de um banco de dados é o processo que permite armazenar os dados em alguma mídia apropriada controlada pelo SGBD. A manipulação inclui funções como: pesquisar em banco de dados para recuperar um dado específico, atualizar o banco para refletir as mudanças no minimundo e gerar relatórios dos dados.

O compartilhamento permite aos múltiplos usuários acessarem, de forma concorrente (simultânea), o banco de dados.

Veja a seguir alguns SGBDs existentes no mercado:



Veja algumas funções básicas do SGBD:

1- Métodos de acesso:

DDL – Linguagem de Definição de Dados;
DML – Linguagem de Manipulação de Dados;
DQL Data Query Language - Linguagem de Consulta de dados.

2- Restrições de Integridade;

3- Segurança de Dados;

4- Controle de Concorrência;

5- Independência de Dados.

Vamos entender cada uma dessas funções:

Métodos de Acesso:

Linguagem de Definição de Dados

Linguagem de definição de dados (LDD ou DDL, do Inglês Data Definition Language) é utilizada para a definição de estrutura de dados.

Exemplo:

Nome	Tipo	Tamanho
Aluno	Alfanumérico	50
RG	Numérico	10
Mensalidade	Moeda	-

A definição do nome, tipo de dados, tamanho e formatos de cada campo, consiste na Linguagem de Definição de Dados (DDL).

A Linguagem de Definição de Dados (DDL) compõe o **Esquema** do Banco de Dados, abrangendo os comandos CREATE, ALTER e DROP. Obs: o DQL inclui o SELECT. Estudaremos esses conceitos mais adiante.

Linguagem de Manipulação de Dados

Também conhecida como DML, do inglês Data Manipulation Language, a linguagem de manipulação de dados consiste em um conjunto de comandos que permite a atualização de informações do banco, como a inclusão e a remoção de informações. A DML interage diretamente com os dados dentro das tabelas. Os bancos de dados mudam à medida que informações são inseridas ou apagadas.

A coleção de informações armazenadas no banco de dados em um determinado momento é chamada de **instância** do banco de dados:

Exemplo:

Cadastro de Alunos	
Nome:	Antônio Neves
RG:	57.777.777-7
Valor da Mensalidade:	R\$ 1.876,00

Instância

Então, ao realizar um cadastro em um aplicativo, por exemplo, dizemos que os dados são “instanciados”, ou seja, inseridos no BD.

A Linguagem de Manipulação de Dados (DML) compõe a Instância do Banco de Dados, abrangendo os comandos INSERT, UPDATE e DELETE.

Linguagem de Consulta de Dados

Também é conhecida como DQL, do inglês Data Query Language, a Linguagem de Consulta de Dados tem como principal instrução SQL o **select**, comando que realiza consultas no SGBD. O conceito será melhor estudado em unidades futuras.

Restrição de Integridade:

Você já ouviu falar sobre uma pessoa íntegra?

O que significa?

Uma pessoa íntegra é uma pessoa honesta, imparcial, correta!

Bom, então você já pode imaginar que integridade de dados tem por objetivo garantir que as informações no banco sejam corretas e confiáveis!

As três formas mais comuns são a integridade são: de **domínio**, de **entidade** e **referencial**.

A **Integridade de Domínio** é responsável por verificar se os valores inseridos no banco de dados, por meio de atributos, estão corretos de acordo com as regras de validação estabelecidas.

Por exemplo: Imagine um formulário para preenchimento que peça o número do telefone e o usuário escreva um texto nesse campo ao invés de números, ou ainda, que peça o número de filhos e o usuário digite -1 (Menos 1 filho???).

Ou imagine um formulário de cadastro de produto em que o usuário digite a data de validade menor que a data de fabricação desse produto!

Cadastro de Requerente

Tipo do Requerente

Pessoa Física

CPF

Nome

Endereço

Cep

UF

Goiás

Município

Selecione

Telefone Residencial

Telefone Celular

Email (será utilizado como login)

Senha

Repita a Senha

Salvar

Para tratar esses problemas, é muito comum usar regras de validação.

Essas regras impedem que o sistema aceite valores fora da faixa de permissão e não permitem as inclusões sem a presença de todos os atributos obrigatórios.! Você estudará isso mais adiante!

A **integridade de Entidade** valida os dados permitidos, a partir dos valores que já foram inseridos no banco e permite a inserção de novos registros apenas após esta verificação.

Exemplos:

- Impedir o empréstimo de um livro que já esteja reservado para outro aluno.
- Não permitir que mais de um aluno seja cadastrado com o mesmo número de matrícula.



Ainda no contexto escolar: sistemas que utilizam validações de integridade de entidade não permitem que mais de um aluno seja cadastrado com o mesmo número de matrícula.



A **integridade referencial** é responsável por verificar se as operações em bancos atendem às regras de relacionamento que já foram definidas para as tabelas do banco de dados. Garante que haja consistência entre os registros no Banco de Dados.



Por exemplo: Imagine que em um Banco de Dados de um cartório, fosse apagado o nome de um homem sem saber se ele tem filhos ou não. Nessa exclusão indevida, o cartório perderia os dados do pai e, conseqüentemente, os dos filhos que estariam relacionados a ele.

Segurança de Dados:

Um SGBD deve manter um sistema de segurança que protege a base de dados de acessos não autorizados. Para isso, são impostas regras de confiabilidade que visam certificar que somente pessoas autorizadas terão acesso a determinadas informações.

Segundo Sêmola (2003, p. 45), “*toda informação deve ser protegida de acordo com o grau de sigilo de seu conteúdo, visando à limitação de seu acesso e uso apenas pelas pessoas para quem elas são destinadas*”



O sistema de segurança do SGBD também deve garantir que a informação mantenha todas as características originais. Para isso são impostas regras de integridade que definem quais pessoas podem ter acesso à base de dados e quais operações podem realizar. Por exemplo, um cliente pode consultar o extrato da sua conta bancária, mas não pode alterá-lo.

O SGBD deve dispor de procedimentos de segurança física e lógica que permitem efetuar cópias e recuperação de dados em caso de falhas, de modo a garantir a segurança e a integridade dos dados.

- Segurança física: proteção dos dados contra roubo, destruição mal-intencionada, atualização não autorizada, incêndios, desastres naturais, furto, entre outros.
- Segurança lógica: proteção por meio de dados criptografados, como senhas (passwords), assinatura digital, por exemplo.

Controle de Concorrência:

Controle de concorrência deve ser utilizado para fazer o controle de transações que são acessadas ao mesmo tempo por usuários distintos que acessam a mesma informação em um banco de dados concomitantemente.

Um SGBD multiusuário deve permitir que vários usuários possam acessar os dados ao mesmo tempo. O controle de concorrência deve evitar o acesso desordenado ao Banco de Dados, fazendo com que os resultados das atualizações sejam sempre corretos.

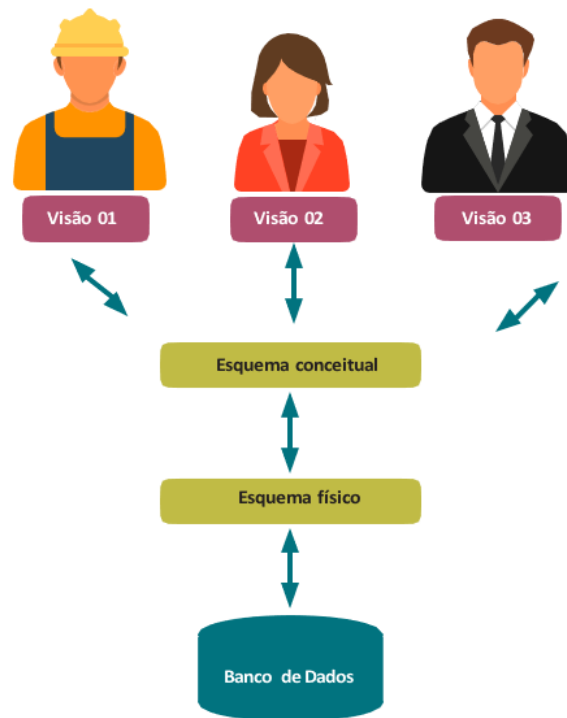
Imagine que você tenha uma conta bancária conjunta com alguém e que vocês estão fazendo um saque simultâneo do total do saldo, em caixas eletrônicos distintos. Se o SGBD do banco não tivesse um sistema de controle de concorrência, permitiria que vocês sacassem o dobro do seu saldo, mesmo não tendo limite de crédito.



Independência de Dados:

É a capacidade de conseguir isolar o usuário de detalhes mais internos do Banco de Dados (abstração de dados) e prover independência de dados às aplicações (estrutura física de armazenamento e estratégia de acesso).

Quando um usuário utiliza um Banco de Dados, ele não faz ideia da quantidade de informações que ele está manipulando. Isto porque, o SGBD oculta certos detalhes de como os dados são armazenados ou mantidos, proporcionando ao usuário uma visão abstrata dos dados.



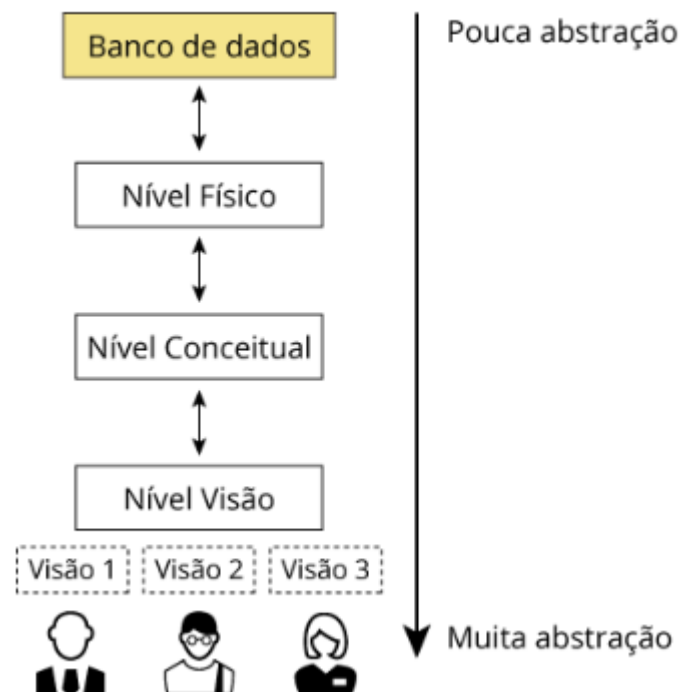
Os Bancos de Dados são frequentemente utilizados por pessoas sem conhecimentos sobre sistemas da informação, então, é preciso que o SGBD manipule dados complexos e apresente ao usuário, as elas com dados referentes ao seu interesse e permissão de acesso, sem que o usuário precise se preocupar.

Para isto, temos três níveis de abstração.

Nível Físico - Nível mais baixo de abstração. Representa com detalhes as estruturas complexas de baixo nível. Em outras palavras, descreve como os dados são armazenados.

Nível Conceitual - Descreve quais dados são realmente armazenados no Banco de Dados e o relacionamento entre esses dados.

Nível Visão - Nível mais alto da abstração no qual se expõem apenas parte do BD. Podem existir diferentes visões para um mesmo banco de dados. São as “telas” que você enxerga quando acessa um determinado sistema. Por exemplo, quando você acessa a tela de extrato da sua conta bancária, sem ter que se preocupar com a forma como as informações foram armazenadas e manipuladas.



Modelo de banco de dados é uma descrição dos tipos de informações que estão armazenadas em um banco de dados. Por exemplo, pode informar que o banco armazena informações sobre produtos e que, para cada produto, são armazenados dados como: o código, o preço e a descrição.

Para construir um modelo de dados, usa-se uma linguagem de modelagem de dados. Existem linguagens textuais e linguagens gráficas e é possível descrever os modelos em diferentes níveis de abstração e com diferentes objetivos.

A seguir, veja os três modelos de Banco de Dados:

Modelo Conceitual:

É a fase em que se busca representar o mundo real por meio de uma visão simplificada dos dados e relacionamentos. Nesse modelo são determinadas **quais** informações serão armazenadas no Banco de Dados, sem se preocupar com o Sistema Gerenciador de Banco de Dados (SGBD) que será utilizado.

Por exemplo:

Cadastro de um Cliente de uma loja – Quais são os dados necessários para esse cadastro?

CPF do Cliente, Nome, endereço e demais dados importantes do cliente para a loja.

Cadastro de uma Compra realizada pelo Cliente de uma loja – Quais são os dados necessários para esse cadastro?

Código da Compra, Data, Quantidade, Preço e demais dados importantes da compra que o cliente realizou na loja.

Nesse nível é possível descrever os tipos de dados requeridos, os relacionamentos entre si e as regras de consistência.



Descreva um modelo conceitual para armazenar os dados de um Produto.

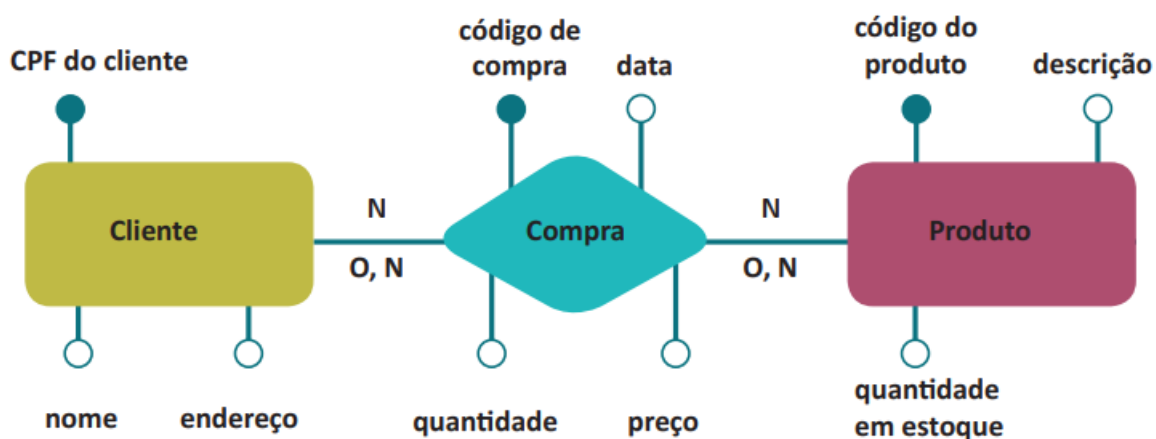
Veja uma provável resposta:

Cadastro de Produto: código do produto, descrição, quantidade em estoque, ...

Modelo Lógico:

Esse modelo compreende uma descrição das estruturas que serão armazenadas no banco e que resulta numa representação gráfica dos dados, de uma maneira lógica, inclusive nomeando os componentes e ações que exercem uns sobre os outros. Nesse nível, o projeto ainda é independente do SGBD e posteriormente poderá ser adequado ao SGBD que for escolhido.

Exemplo de Modelo Lógico:



Neste exemplo, o modelo lógico está representado pelo Diagrama Entidade e Relacionamento (DER), que veremos nas próximas agendas. Neste modelo já é possível compreender como os dados se comportarão no sistema a ser desenvolvido. Analisando o modelo, temos a informação de que o **cliente** tem um **CPF**, um **nome** e um **endereço** e que esse **cliente** pode **comprar produtos**. Pela informação da letra **N**, representada entre os relacionamentos, identifico que um cliente pode comprar **muitos** produtos e que um tipo de produto pode ser comprado por **muitos** clientes.

O modelo lógico também pode ser representado de forma textual.

Mas não se preocupe! Estudaremos o Diagrama Entidade e Relacionamento nas próximas agendas!

Modelo Físico:

A partir da definição de um modelo lógico, desenvolveremos um modelo físico onde serão detalhados os componentes de estrutura física do banco de dados, incluindo tabelas, campos, tipos de valores, regras de restrições etc. Nesse estágio estamos prontos para criar o banco de dados propriamente dito, usando o SGBD preferido.

Exemplo de um Modelo Físico:

PRODUTO								
nome	tipo de dados	tamanho	obrigatório	único	chave primária	chave estrangeira	valor default	regra de validação
codigo_produto	INTEGER	5	sim	sim	sim	não	não	não
descricao	VARCHAR	40	sim	não	não	não	não	não
quantidade_estoque	DECIMAL	10,1	sim	não	não	não	0	valor >=0

Para você entender melhor, veja um exemplo de criação de tabela, a partir do exemplo do Modelo Físico acima, usando o SGBD MySQL para a criação da tabela produto:

```
Create Table Produto (  
codigo_produto INTEGER NOT NULL PRIMARY KEY,  
descricao VARCHAR (40) NOT NULL,  
quantidade_estoque DECIMAL (10,2) NOT NULL)
```

Não se preocupe com esses tipos de projetos porque estudaremos mais adiante!

Até aqui relacionamos as definições, os exemplos e as funções que um Sistema Gerenciador de Banco de Dados executa, mas...



***Quando não devemos utilizar um SGBD?
Sempre que uma empresa ou um cliente tiver um sistema precisará desenvolver um SGBD?***

Apesar de todas as informações colocadas até o momento, existem algumas situações em que o SGBD pode envolver custos adicionais desnecessários. Segue alguns exemplos:

- Alto investimento inicial em hardware, software e treinamento;
- Necessidade de um hardware adicional;
- Banco de Dados simples e bem definido;
- Uso somente de acesso centralizado;
- Esforço adicional para oferecer funções de segurança, controle, recuperação e integridade de dados.

Características de um Banco de Dados

Você já conhece a definição de um Banco de Dados, então agora vamos estudar algumas de suas características:

- 1- Um BD é projetado, construído e “preenchido” com dados para um propósito específico. Existe sempre um grupo de usuários e um conjunto de aplicações pré-estabelecidas que vão trabalhar em torno de um BD.

Por exemplo, você tem uma ideia de como será o BD e, então, começa a imaginar, pensar como ele deve ser e, em seguida, começa a modelá-lo.



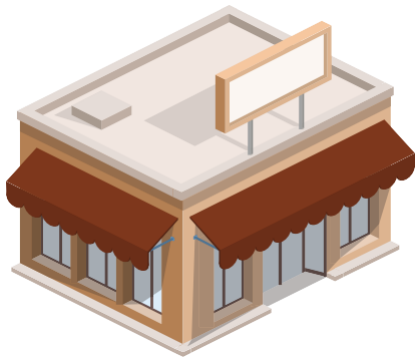
Modelar vem de modelo. Modelo é um protótipo, em escala menor do produto que queremos implementar ou da solução que desejamos obter. O Modelo Dados nos permite criar, testar funcionalidades novas e avaliar o projeto com um baixo custo, antes de sua implementação.

- 2- Um BD representa algum aspecto do mundo real. Toda mudança realizada no mundo real deverá ser “levada” ao Banco de Dados.

Por exemplo, se você pensar em um projeto de uma casa, precisará idealizar seu Banco de Dados com todas as informações necessárias, como por exemplo: quais os cômodos que a casa terá, qual a metragem, o tipo de casa (térrea ou sobrado) etc..

- 3- Um BD pode variar em tamanho e complexidade (conforme a situação, problema e a empresa ou cliente).

Por exemplo, o Banco de Dados da Receita Federal é muito maior em tamanho e em complexidade comparado com o Banco de Dados de uma microempresa, não é?



4- Um BD deve ter controle de redundância, ou seja, não ter dados em duplicidade. No processamento de Dados Tradicional, cada grupo de usuários mantém seus próprios arquivos. Isto leva ao armazenamento múltiplo dos mesmos dados, causando problemas de espaço e inconsistência dos dados. Sendo assim, ocorrem redundâncias que prejudicam os sistemas, tais como:

- ✓ Toda vez que for necessário atualizar um arquivo de um grupo, é necessário atualizar todos os grupos para manter a integridade dos dados no ambiente como um todo;
- ✓ Redundância desnecessária de dados acaba levando ao armazenamento excessivo de informações, ocupando um espaço que poderia ser utilizado com outras informações.



5- Um BD deve compartilhar os dados, ou seja, deve permitir que vários usuários acessem os dados ao mesmo tempo.

Por exemplo, ao acessar uma rede social, você e seus colegas podem manipular os mesmos dados simultaneamente sem problema algum.

Atenção! Conforme já foi comentado, o SGBD deve incluir um software de controle de concorrência para evitar o acesso desordenado ao BD, fazendo com que os resultados das atualizações sejam sempre corretos.



6- O BD deve manter um padrão em nível de elementos de dados, telas, estruturas de documentos entre outros. É mais fácil manter a padronização em um BD centralizado do que em um ambiente onde cada grupo de usuário mantém controle sobre seus arquivos e softwares.

Mais à frente estudaremos modelos de classes, nos quais os atributos descritos nas classes podem se transformar em colunas de uma tabela que podem gerar confusão quando não for adotada uma determinada padronização.

Observe que o atributo “Nome”, da imagem a seguir, não especifica a que se refere, podendo gerar dúvidas por não ser possível afirmar se “Nome” é o nome do cliente, do produto, do fornecedor, do atendente ou qualquer outro elemento.

NotaFiscal
ID
Data
Valor
Descricao
Nome

A padronização dos nomes dos campos, definida pelo sufixo “Cli” para todos os campos pertencentes ao cliente esclareceria essa dúvida.

NotaFiscal
ID_Nota
Data_Nota
Valor_Nota
Descricao_Prod
Nome_Cli

Existem vários outros exemplos que não são necessariamente derivados de modelos de classes. Há casos de tabelas sem significado aparente, com nomes indecifráveis como “XX”, “tblTeste”, “tbl_001”, “Plan1” etc, cujas colunas seguem com os mesmos problemas de entendimento. Na grande maioria das vezes, não tem o comentário preenchido, dificultando ainda mais o entendimento.

(Adaptado de DevMedia, disponível em <https://www.devmedia.com.br/padronizacao-de-nomenclatura-revista-sql-magazine-100/24710>. Acessado em 13/09/2018).

Estas características que acabamos de elencar são as principais e as fundamentais para que o SGBD e o BD trabalhem juntos em harmonia. Porém, sempre surge uma dúvida: quem são as pessoas responsáveis por gerenciar, projetar e manipular o Banco de Dados?

Os usuários de Banco de dados são aqueles que interagem direta ou indiretamente com o SGBD. São eles:

a) Administrador de Banco de Dados (DBA - Database Manager)

Um grande Banco de Dados precisa de uma grande equipe trabalhando para que ele se mantenha íntegro e coeso. Para isso, é necessário uma pessoa para **manter o controle central dos dados** e dos programas que acessam esses dados, o chamado DBA, também responsável pela concessão de autorização para acesso aos dados.



b) Projetista de Banco de Dados (DBD- Database Designer)

Este profissional tem como função principal identificar os dados a serem armazenados e definir a estrutura apropriada para representar e armazenar estes dados. **Este deve interagir com os usuários para descobrir suas necessidades e encontrar a solução ideal para as mesmas.**

O projeto final do Banco de Dados deve ser capaz **atender as necessidades de todos os grupos de usuários do mesmo.**



c) **Usuários finais**

São divididos em alguns grupos:

- **Usuários Ocasionais:** são usuários habituados ao uso do SGBD e que não utilizam programas, elaborando suas consultas diretamente em linguagens de consultas.
- **Usuários simples:** são usuários que precisam dos programas de aplicações para interagir com o sistema.
- **Usuários especializados:** são usuários que elaboram sistemas altamente especializados, interagindo indiretamente com o SGBD.

