

Curso Técnico em Desenvolvimento de Sistemas

**Introdução ao Desenvolvimento de Sistemas**

 **INTRODUÇÃO A**

**BANCO DE DADOS**

##### A necessidade de organizar informações acompanha a humanidade desde o início dos tempos. O pastor representava ovelhas por meio de pedras, enquanto os escribas ordenavam os textos nas estantes e a bibliotecária organizava, manualmente, em ordem alfabética, as revistas e livros nas estantes.



***INTRODUÇÃO A BANCO DE DADOS***

***Acontece que a população mundial cresceu, assim como a quantidade de informações disponíveis e, desse modo, pesquisá-las tornou-se mais complexo, o que exigiu novas formas de organização.***

Imagem 1- Informações organizadas. A imagem apresenta informações organizadas em calendários, gráficos, e-mails, celular, entre outros.

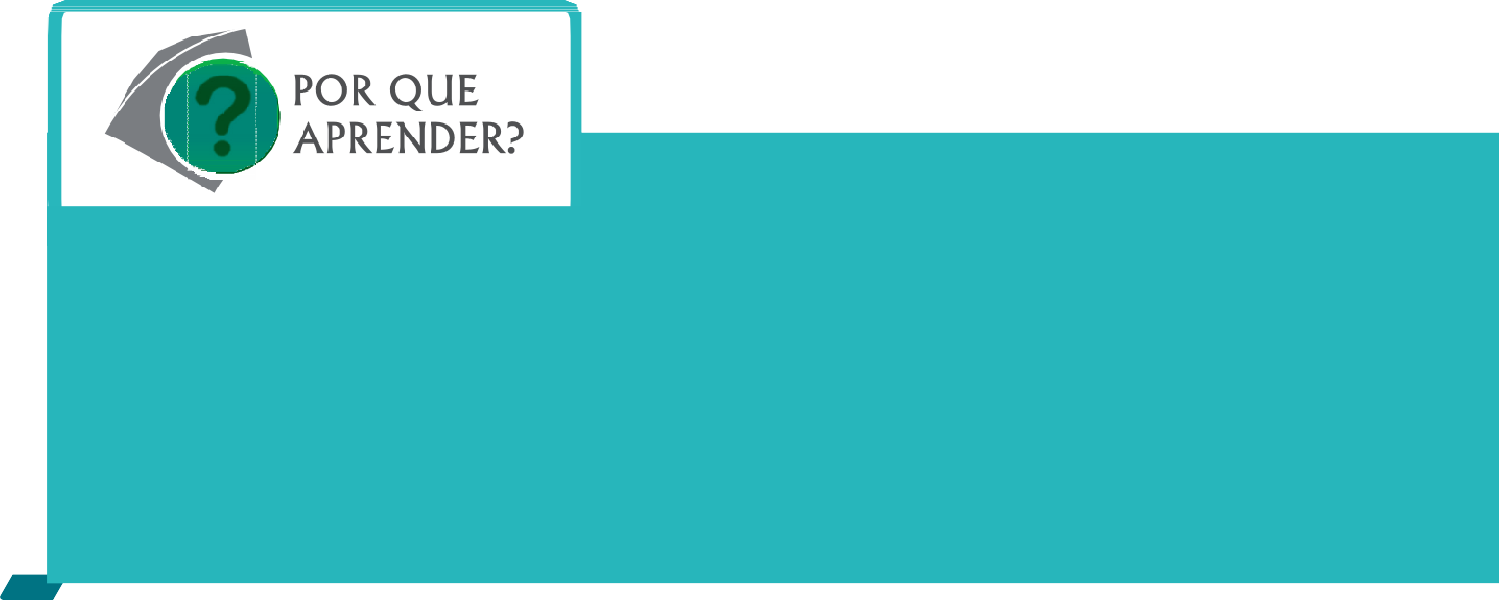
##### Tente encontrar um livro na biblioteca municipal sem saber como estão organizadas

***as estantes. Vá procurando estante por estante, livro por livro... Quanto tempo irá demorar? Agora, pense em fazer a mesma pesquisa na Biblioteca Nacional (do Rio de Janeiro) ou na Biblioteca do Smithsonian Museum, dos Estados Unidos. Sem uma organização prévia, fica muito demorado e trabalhoso obter as informações de que precisamos em nosso dia a dia.***

***E como a informática pode ajudar na organização? Muito! Em todo esse processo, que inclui armazenamento, manutenção e consulta de informações, o Banco de Dados proporciona agi- lidade, uniformidade e segurança em todas as suas fases. Ao juntarmos o conhecimento pre- existente à velocidade de processamento e à capacidade de armazenamento de informações, chegamos a modelos interessantes no que diz respeito à facilidade de uso, velocidade de acesso e de respostas, além de baixo custo de manutenção. Este é o objetivo da existência***

***do Banco de Dados que iremos abordar nesta agenda. Então, vamos começar?***

*Adaptado de Piva, Gustavo Dibbern Informática, análise e gerenciamento de dados, 2010. Págs. 50 e 51.*



?

Um conjunto de dados pode gerar informações valiosas para as empresas. O Banco de Dados pode ajudar uma organização a tomar decisões estratégicas conhecendo seu público e construindo bons relacionamentos com seus clientes, além de auxiliar e otimizar os processos operacionais da empresa, como: contabilidade, gestão de projetos, recursos humanos e muitas outras informações.

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

*5*

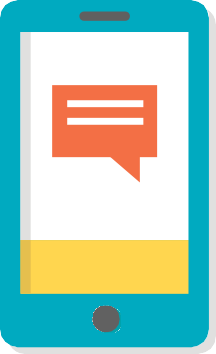
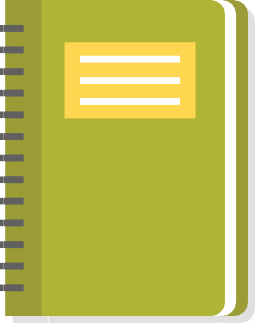
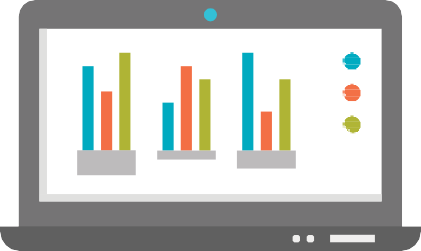


A

Já falamos um pouco sobre Banco de Dados nessa agenda, mas será que você sabe a definição desse termo? Va- mos lá:

Basicamente, podemos definir Banco de Dados como sendo um conjunto de informações inter-relacionadas sobre determinado assunto e armazenadas de forma a permitir acesso organizado por parte do usuário.

“Um banco de dados representa algum aspecto do mundo real, às vezes chamado de minimundo ou de universo de discurso (UoD – Universe of Discourse).” (Elmasri; Navathe, 2011) - Elmasri, Ramez; Navathe, Shamkant B. (2011) Sistemas de Bancos de Dados. Pearson, 6a. edição em português.





VOCÊ NO COMANDO

***Existem diversos exemplos de Banco de Dados no nosso dia a dia. Pare, pense e relacione a definição citada acima com mais alguns outros exemplos.***

Imagem 2- Meios usados para organizar as informações. A imagem apresenta diversas formas para se organizar as informações representadas em um notebook, caderno, e celular.

Ter uma massa de informações desorganizadas na qual não se consegue encontrar o que se quer é inútil, é o mesmo que ter nada. Quantos de nós conseguimos nos organizar em casa? Buscar ex- atamente a foto que queremos ver, o número do telefone do disk- pizza, o manual da máquina de lavar louças que está quebrada, um livro que você comprou e ainda não leu, um artigo de revista que você precisa reler? E uma outra pergunta é: quanto tempo você precisa para encontrá-los?

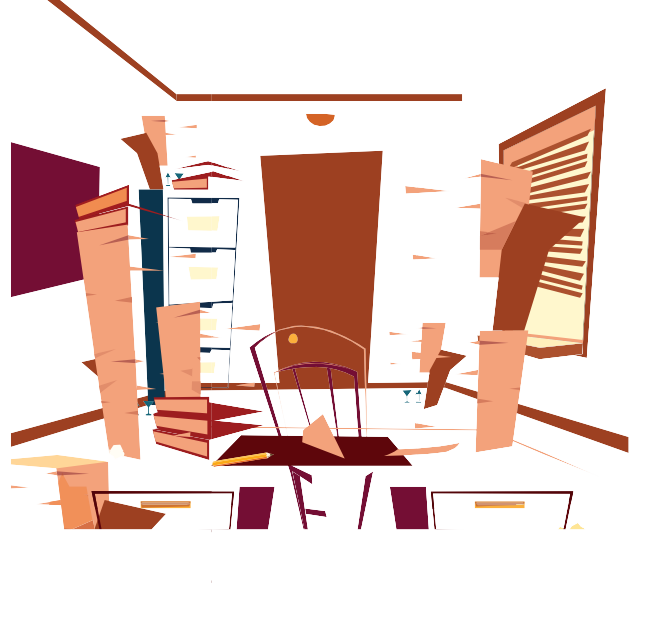


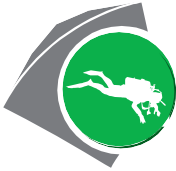
Imagem 3- Pilha de papéis e caixas. A imagem apresenta informações desorganizadas representada por pilhas de caixas e papéis em uma mesa.

Tudo isso também pode acontecer no Ambiente Virtual. Quando ligamos o computador, lá estão milhares de arquivos: apresen- tações, textos, músicas, vídeos, planilhas etc.aumentam de quantidade muito rápido e facilmente podem tornar-se difíceis de localizar. Esse problema acontece também nos ambientes corpo- rativos, onde as informações precisam ser coletadas e organizadas

*6*

para que possam ser utilizadas a fim de auxiliar nas tomadas de decisões e nos processos operacionais da em- presa.

Para entender como trabalhar com as informações para que sejam facilmente organizadas e recuperadas, vamos mergulhar no tema dessa agenda!

Adaptado de Alessandro Nicoli de Mattos. Informação é prata. Compreensão é ouro, 2010. Disponível em [**http://plus.google.com/118410405608511378703**.Acessado](http://plus.google.com/118410405608511378703.Acessado) em 11/09/2018.



Já que sabemos a definição e vimos exemplos de Banco de **Dados**, uma pequena dúvida: Por que chamamos Banco de Dados? Não

poderia ser Banco de **Informações**?

Imagem 4- Dois dados. A imagem apresenta dois dados usados

em jogos representando o conceito de banco de dados.

m

m

Local que será arma- zenado os dados. Conhecido também como colunas.

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 5 a seguir:

Forma gráfica de apre- sentar os campos com os dados.

Um SGBD é um pacote deferramentas

com um propósito geral que facilita os processos de definição, construção e manipulação de Banco de Dados para

qualquer aplicação. Ex. SQL Server

BD - Banco de Dados

Tabelas

Campos

SGBD – Sistema

Gerenciador de

Banco de Dados.

Dados

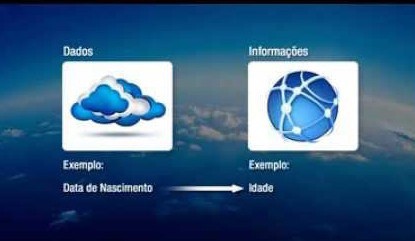
Imagem 5- Estrutura de umbanco de dados. A imagem apresenta uma pirâmide apresentando a

estrutura de um banco de dados representada por tabelas, campos e dados

*7*

Assista ao vídeo que traz os primeiros conceitos de Banco de Dados. Disponível em

**https:**[**//www.youtube.com/watch?v=hap3zFj3zSg**.](http://www.youtube.com/watch?v=hap3zFj3zSg)



A estrutura de Banco de Dados ajuda 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)**

Um Sistema Gerenciador de Banco de Dados (SGBD) é um pacote de ferramentas com um propósito geral que facilita os processos de definição, construção, compartilhamento e manipulação de 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 pesquisas em banco de dados para recuperar um dado especí- fico, atualização do banco para refletir as mudanças no minimundo e gerar os relatórios dos dados. O

Imagem 6-Logotipos de banco de dados. A imagem apresenta algunslogos de banco de dados existentes no mercado: Mysql, postgreSQL, Oracle.

compartilhamento permite aos múltiplos usuários e acessarem,

de forma concorrente, o banco de dados.

A Figura acima exemplifica alguns SGBDs e xistentes no mercado de trabalho. Veja algumas funções básicas do SGBD:

* Métodos deacesso:
* DDL – Linguagem de Definição deDados;
* DML – Linguagem de Manipulação deDados;
* DQL Data Query Language - Linguagem de Consulta dedados.
* Restrições deIntegridade;
* Segurança deDados;
* Controle deConcorrência;
* Independência deDados.

Vamos entender cada uma dessas funções:

**Métodos de Acesso:**

**nguagem de Definição deDados**

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

A manipulação de Dados consiste em:

Remoção de informações

Inserção de informações

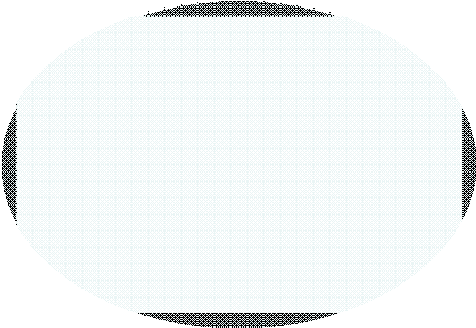
* Consultas;
* Atualizações

A linguagem de manipulação de dados (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:

Instância



**Cadastro de Alunos**

Nome:

RG:

Valor da Mensalidade: R$ 1.876,00

57.777.777-7

AntônioNeves

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

#### Restrição deIntegridade

Você já ouviu falar sobre uma pessoa íntegra? O que significa?

Uma pessoa íntegra é uma pessoa honesta, im- parcial, correta!

Bom, então você já pode imaginar que integri- dade de dados tem por objetivo garantir que as informaçõesnobancosejamcorretaseconfiáveis!

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

A **Integridade de Domínio** é responsável porveri- ficarseosvaloresinseridosnobancodedados,por meiodeatributos,estãocorretosdeacordocomas regras de validaçãoestabelecidas.

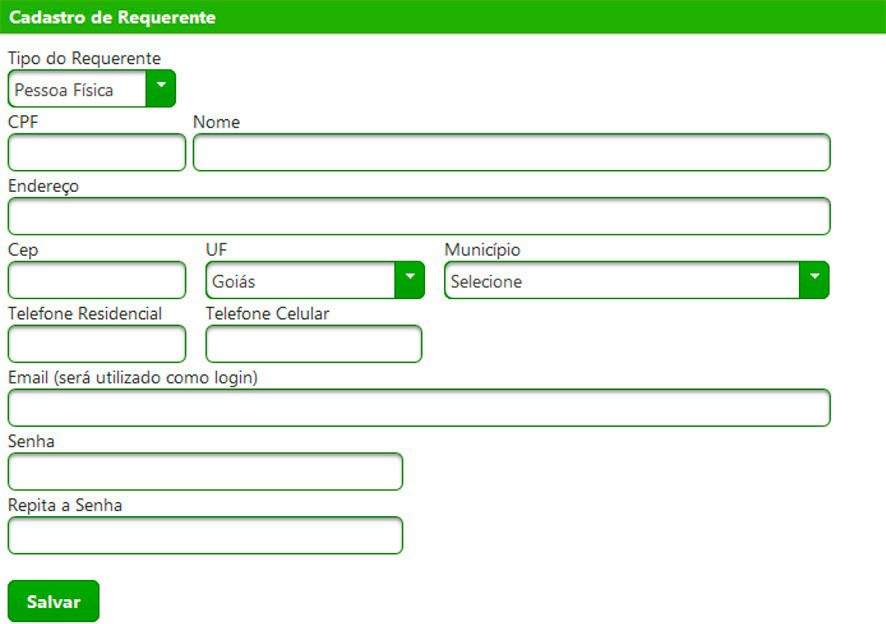


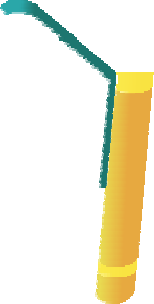
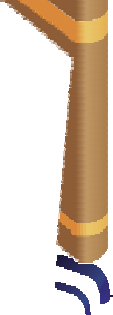
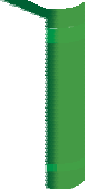
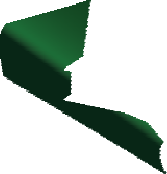
Imagem 7- Tela de um sistema. A imagem apresenta a tela de cadastro de um sistema qualquer representado pelos campos: CPF, nome, endereço, CEP, UF, Município, telefone residencial e celular, e-mail entre outros.

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 desseproduto!

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



inseridosnobancoepermiteainserçãodenovosregistrosapenasapósestaverificação.

Exemplos:

* Impediroempréstimodeum livroquejáestejareservado para outro aluno.
* Não permitir que mais de um aluno seja cadastrado com omesmo número de matrícula.

Imagem 8- Livros. A imagem apresenta uma pilha de

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

Imagem 9 - Um grupo de pessoas. A imagem apresenta um grupo de pessoas em uma hierarquia representado a integridade referencial dos 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, consequentemente, os dos filhos que estariam relacionados a ele.

#### Segurança deDados



Um SGBD deve manter um sistema de segurança que pro- tege 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 pes- soas 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

Imagem 10- Tela de computador. A imagem apresenta a tela de um computador representando a preocupação na proteção e segurança de dados informatizados.

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**

O 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 pos- sam acessar os dados ao mesmo tempo. O controle de concorrên- cia 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

Imagem 11- Caixa eletrônico

. A

imagem apresenta uma pessoa realizand o transaçõe s no caixa eletrônico de um banco.

ODUÇÃO A BANCO DE DADOS

***R***

***INT***

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 deDados

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, proporcion- ando ao usuário uma visão abstrata dos dados.



**Visão 01**



**Visão 02**



**Visão 03**

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 apre- sente ao usuário, as telas com dados referentes ao seu interesse e permissão de acesso, sem que o usuário precise se preocupar.

**Esquemaconceitual**

**Esquemafísico**

**Banco deDados**

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.Sãoas“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.

Imagem 12-

Níveis de abstraçã

o. A imagem apresent a os três níveis de abstraçã o em um banco de dados que consiste em: nível visão, nível físico e nível conceitu al.

#### Modelos de Banco de Dados

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.



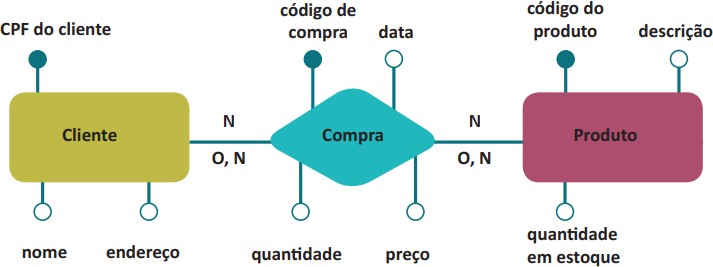
Descreva um modelo conceitual para armazenar os dados de um Produto. Veja uma provável resposta:

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 **comprarprodutos**. 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.

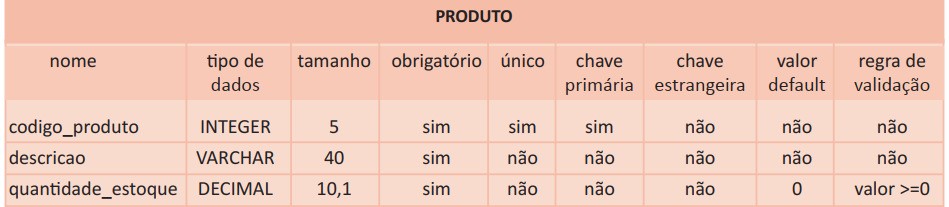
O modelo lógico também pode ser representado de forma textual.

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

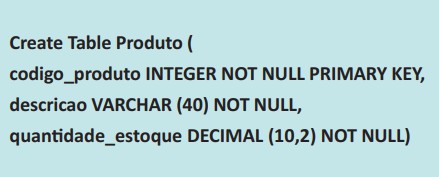
Modelo Físico:

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:



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:



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.***

1. 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..

1. 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?

Imagem 12- Microempresa e Receita Federal. A imagem apresenta uma microempresa e o símbolo usado pela Receita Federal

1. 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 do sdados 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.

Imagem 13- Uma pasta. A imagem apresenta uma pasta com dados representando a importância de controlar esses dados para evitar a redundância.

1. 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.

1. 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 esoftwares.

Imagem 14-

Celular. A

imagem apresent a duas pessoas comparti lhando dados por meio do celular.

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 dúvi- das por não ser possível afirmar se “Nome” é o nome do cliente, do produto, do fornecedor, do atendente ou qualquer outro elemento.

ID

Data Valor Descricao Nome

NotaFiscal

ID\_Nota Data\_Nota Valor\_Nota Descricao\_Prod Nome\_Cli

NotaFiscal

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

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 temo comen- tário preenchido, dificultando ainda mais oentendimento.

(Adaptado de DevMedia, disponível em

**https://**[**www.devmedia.com.br/padronizacao-de-nomenclatura-revista-sql-magazine-100/24710**.](http://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 ger- enciar, projetar e manipular o Banco deDados?

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

#### Administrador de Banco de Dados (DBA - DatabaseManager)

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

#### Projetista de Banco de Dados (DBD- DatabaseDesigner)

Este profissional tem como função principal identificar os dados a serem armazenados e definir a estrutura apropriada para representar e armaze- nar 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 necessi- dades de todos os grupos de usuários do mesmo.**

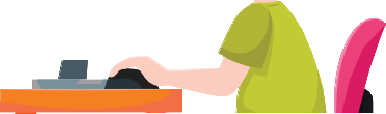
Imagem 16-

Cadeado. A imagem apresenta um cadeado representa ndo uma forma de manter os dados em segurança.

Imagem 17- Projeto de banco de dados. A imagem mostra uma pessoa em uma escada representando um projetista de banco de dados.



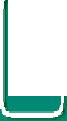
#### Usuáriosfinais



São divididos em alguns grupos:

* Usuários Ocasionais: são usuários habituados ao uso do SGBD e que nãoutilizam 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.

Imagem 18- Uma pessoa. A imagem apresenta um usuário comum sentado na frente de um computador executando suas atividades.



Estes exercícios devem ser entregues de forma on-line como atividades da agenda.



#### Exercício 1

Em uma Clínica Veterinária são usados dois cartões de controle de vacinação para cada animal vacinado, ficando um com o veterinário, que aplica as vacinas e ooutro com o atendente, que controla o pagamento.

Certo dia, o atendente foi ao banco enquanto o veterinário vacinou um animal. No mês seguinte, devido às divergências entre as duas fichas (a do atendente e a do veterinário), a clínica ficou em dúvida a respeito da vacinação do animal em questão. O fato acima exemplifica um problema de:

1. Redundância dedados.
2. Inconsistência dedados.
3. Definição de Banco dedados.
4. Todas asalternativas.
5. Nenhuma dasalternativas.

#### Exercício 2

Emumcanhotodetalãodecheques,sãoanotados,paratodochequeemitido,adatadeemissão,ovaloreo destino do cheque. Cada um desses itensconstitui:

1. Dado.
2. Meta dedados.
3. Banco dedados.
4. Sistema Gerenciador de Banco dedados.
5. DBA.

#### Exercício 3

A loja X desenvolveu uma lista para controlar seus clientes inadimplentes, logicamente, os dados constantes nesta lista eram os nomes e registros dos maus pagadores. A lista foi distribuída aos vendedores da loja, para que estes estivessem cientes dos nomes constantes nela,sendo orientados que a cada “inadimplente” que acer- tasse sua situação junto à loja, tivesse seu nome marcado. Posteriormente, a ideia da lista foi copiada por uma grande loja, mas com o mesmo propósito e foram utilizadas ferramentas informatizadas para excluir os nomes da lista, incluir novos nomes e realizar as consultas ao banco. Estas características descrevem:

1. Processamento tradicional dearquivos.
2. Um banco dedados.
3. Abstração dedados.
4. Administrador dedados.
5. Compartilhamento dedados.

#### Exercício 4

Sérgio comanda uma equipe em uma grande organização. É ele quem controla o acesso dos funcionários aos dados da empresa, mantém o controle central e responde por todo problema que possa haver com o banco de dados. Marcelo, por sua vez, subordinado a Sérgio, define quais dados devem constar no banco e suas estruturas e também define como estes dados serão armazenados. Marcelo e Sérgio são respectivamente:

1. DBA eDBD.
2. DBD eDBA.
3. DBD e Usuáriofinal.
4. Usuário final eDBA.
5. Usuário final eDBD.

#### Exercício 5

Um grupo de analistas se reuniu para elaborar um sistema de Folha de Pagamento. Resolveram, então, se organizar em subgrupos para dividir tarefas. O primeiro grupo definiu a estrutura dos dados, o segundo grupo definiu quais dados seriam armazenados no banco. Do trabalho dos dois grupos surgiu a seguinte tabela:

|  |  |  |
| --- | --- | --- |
| Nome do empregado | caractere | 50 |
| Salário | numérico | 10 |
| INSS | numérico | 08 |

O terceiro grupo preencheu todos os campos do banco e manipulou estes dados com consultas, inclusões, alterações e exclusões. Os modelos acima referidos são respectivamente:

* 1. Conceitual, físico evisão.
  2. Conceitual, lógico e físico.
  3. Físico, conceitual evisão.
  4. Físico, lógico econceitual.
  5. Físico, visão econceitual.

Expectativas de Respostas:

1. B
2. A
3. B
4. B
5. B

**AMPLIANDO HORIZONTES**

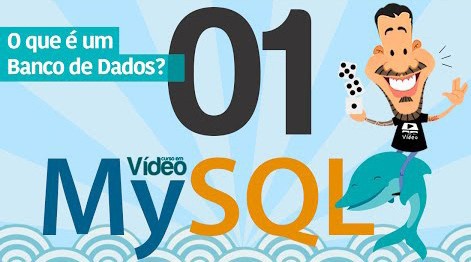


**Vídeo:**

Não deixe também de assistir ao vídeo:

Curso MySQL #01 – O que é um Banco de dados? Saiba como funciona um Banco de Dados e como eles surgi- ram no mundo da tecnologia. Disponível em **https:**[**//www.youtube.com/watch?v=Ofktsne-utM**.](http://www.youtube.com/watch?v=Ofktsne-utM)

Acessado em 13/09/2018.

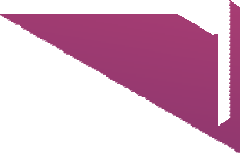
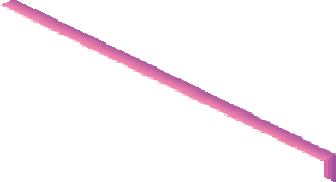
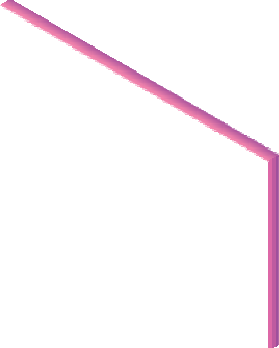
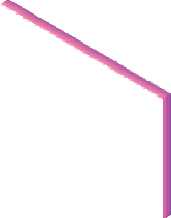
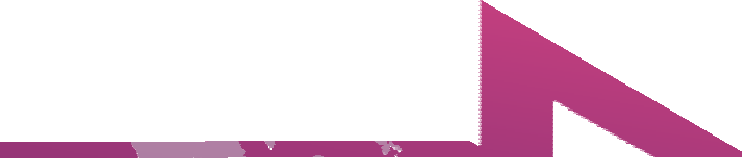


Para aprofundamento dos temas discutidos nesta aula, seguem abaixo algumas dicas de livros e cursos online que se relacionam com o conteúdo estudado. Estas dicas são muito importantes para você!

**Livros:**

ELMASRI, R.; NAVATHE, S. B. Sistemas de bancos de dados. 6 ed. Pearson, 2012. TAKAHASHI, M.; AZUMA, S. Guia mangá de banco de dados. 2 ed. Novatec: 2011. DATE, C.J. Introdução a Sistema de Banco de Dados. 8 ed. Campus, 2013

INTRODUÇÃO A BANCO DE DADOS



**AGENDA09**

**MODELAGEM DE DADOS**

*22*

a

q d

e

##### Você já brincou de massinha? Já fez vários animais, comidas e até mesmo objetos de decoração com elas?



***Quando fazíamos essa brincadeira era justamente para modelar ou imaginar como seria aquele objeto, tentando fazer uma cópia com a máxima perfeição.***

***Os modelos de Banco de Dados têm essa mesma intenção: demonstrar como serão construídas as estruturas de dados que darão suporte aos processos de negócio, como esses dados estarão organizados e quais os relacionamentos que pretendemos estabelecer entre eles.***

Imagem 01: Fonte: https://alimentosaudeinfantil.wordpress.com/category/massi nha-de-modelar-comestivel/. A imagem apresenta uma centopéia colorida feita com massinhas de moldar.



?

A importância dos Bancos de D dos cresceu devido à necessidade das empresas em armazenar grandequantidadedeinformaçãocomrapidez,simplicidadeeconfiabilidade,equ pudessemseraces- sadas a qualquer momento. Mas, para ue isso fosse possível, foi preciso o desenvolvimento de méto- dos para projetar e construir Bancos e Dados. É sobre essa metodologia que trataremos nesta aula: o Modelo de Entidades e Relacionamentos (MER ou Modelo ER) e o Diagrama Entidade Relacionamento (DER)



~~A~~

Você já parou para pensar na quantidade de dados que são necessários para um sistema de controle de estoques de um supermercado? São muitas! Por exemplo, quando você compra uma quantidade de sabão em pó no supermercado, como será que os dados são relacionados para que essa quantidade seja baixada no estoque assim que a compra for registrada no caixa? Interessante, não é?

Entretanto, para que a venda do cliente se relacione com o produto que está no estoque é necessário relacionar esses dados. Pensando nisso, vários autores desenvolveram modelos e técnicas para relacionar e organizar os dados. Chamamos essas técnicas de modelagem de dados.

O Modelo de Entidade e Relacionamento é muito utilizado, devido a sua simplicidade e eficiência e sua finalidade é representar as estruturas de dados da forma mais próxima do mundo real, descrevendo os objetos, as características e as relações entre eles. A principal ferramenta desse modelo é o Diagrama de Entidade e Relacionamento (Diagrama ER ou DER) que traz representações gráficas, seguindo determinadas regras que serão utilizadas na implementação do sistema.

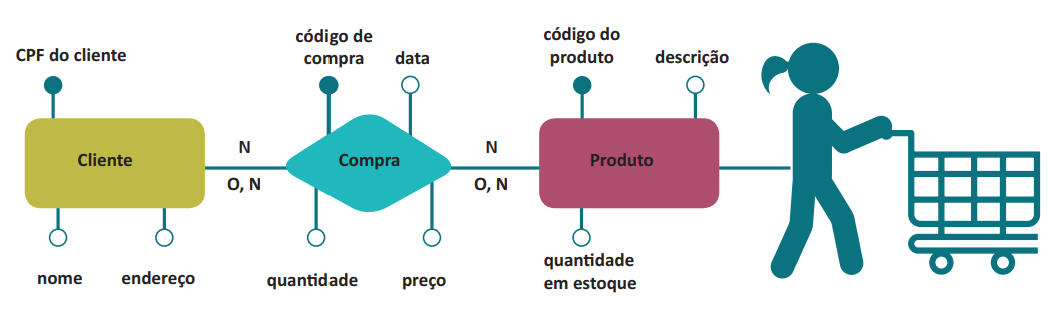


Imagem 02: A imagem apresenta de forma gráfica o Diagrama de Entidade e Relacionamento (Diagrama ER ou DER) representando a relação entre as entidades cliente, compra e produto em um supermercado. A entidade cliente tem como atributos: CPF do cliente, nome e endereço, a entidade compra com os campos: código da compra, data, quantidade e preço e a entidade produto com os campos: código do produto, descrição e quantidade no estoque, após o relacionamento temos uma moça com um carrinho de compras de supermercado.

Há vários componentes do modelo ER. Vale, portanto, conhecer os principais, entre os quais se alinham: Enti- dade, Relacionamento e Atributos.



O Modelo Entidade-Relacionamento é um modelo conceitual usado para identificar como as entidades (pes- soas, objetos ou conceitos) com suas propriedades e

características (atributos) se relacionam entre si dentro de um sistema (relacionamento).

Por meio de um conjunto definido de símbolos, representamos no Diagrama Entidade Relacionamento (DER), a

estrutura do banco de dados do sistema que se pretende desenvolver. Vamos entender os conceitos do modelo ER:

**Entidade:** é a representação de um objeto no mundo real com existência interdependente. Pode ser um objeto

com existência física ou conceitual.

existênciafísica

reserva de avião



ênciaconceitual

exist

Imagem 03: A imagem apresenta entidades do tipo existência física como: uma pessoa, um caminhão e um avião e como existência conceitual uma reserva de avião.

A representação de uma Entidade no Diagrama Entidade-Relacionamento é feita por um retângulo, com o nome desta entidade no singular, em seu interior, conforme imagem 4 a seguir:

pessoa

caminhão

avião

reserva de avião

Imagem 04: A imagem apresenta exemplos de entidades representadas dentro dos retângulos com nomes específicos como:pessoa, caminhão, avião e reserva de avião.

**Relacionamentos:**Dentro de um contexto do mundo real uma entidade raramente apresenta-se isolada, ou seja, sem associação com nenhuma outra.Na maioria das vezes, o que ocorre durante o levantamento de dados, é a identificação de alguma conexão com outra entidade.

A essa conexão lógica entre duas ou mais entidades dá-se o nome de relacionamento e é representado por um losango.

Por exemplo: a entidade “Aluno” está relacionada à entidade “Curso” porque possuem uma associação entre si, ou seja, para saber em qual curso o aluno está matriculado e quais os alunos estão matriculados em determi- nado curso, é preciso existir um relacionamento entre essas duas entidades.



Imagem 05: A imagem apresenta o relacionamento entre as entidades aluno e curso representadas pelo retângulo, e no meio um losango para criar a conexão entre elas.

**Atributos:** são propriedades particulares que descrevem uma determinada entidade. Uma entidade terá valores próprios em seus atributos. Por exemplo, uma entidade “Pessoa” pode ter os seguintes atributos que a descrevem: CPF, nome, telefone e e- mail.



Imagem 06: A imagem mostra a entidade pessoa representada por um retângulo e os seus atributos através de uma elipse para cada atributo: CPF, nome, telefone e email.

O atributo é representado por uma **elipse**.

Agora que você já conhece os componentes do Diagrama ER, vamos detalhar cada um dos tipos dos conceitos

apresentados:

**Tipos de Entidades**

Um BD geralmente contém grupos de entidades similares. Estas entidades possuem os mesmos atributos, embora cada entidade possua os próprios valores. Entidades similares definem um “tipo entidade”, o qual é um conjunto de entidades que possuem o mesmo atributo.

Podemos classificar as entidades de acordo com o sua função dentro do sistema. Assim, podemos classificá- las em Entidade Forte, Fraca ou Associativa. Veja:

Entidade Forte

Uma entidade é considerada forte quando existe por ela só, independentemente da existência de outras entidades. Por exemplo, em um sistema acadêmico, a entidade curso independe de quaisquer outras paraexistir, portanto, é considerada“forte”.

EntidadeFraca

Ao contrário das entidades fortes, as fracas são aquelas que dependem de outras entidades para existir. São identificadas por se relacionarem com entidades específicas. Seguindo o exemplo do sistema acadêmico, a entidade ementa de uma disciplina, só tem sentido se houver a entidade disciplina. Não faz sentido existir uma entidade para armazenar os dados da ementa de uma disciplina que não existe, concorda?

**Ementa**

**possui**

**Disciplina**

Imagem 07: A imagem mostra a entidade ementa representada por um retângulo com traços duplos e um losango com traços duplos, porque ementa é uma entidade fraca, para se relacionar com a entidade disciplina.

Para identificação, a entidade fraca e o seu relacionamento são representados por traços duplos.

EntidadeAssociativa

A entidade associativa é criada quando se identifica a necessidade de associar uma entidade a um relaciona- mento existente. Na modelagem Entidade-Relacionamento não é possível que um relacionamento seja as- sociado a uma entidade, então tornamos esse relacionamento uma entidade associativa para que possamos relacioná-lo com outras entidades.

Para esclarecer melhor esse conceito, vamos usar o exemplo do sistema acadêmico novamente.

Imagine a seguinte situação:

Um professor ministra uma disciplina (aula) em uma data e horário e essa aula é assistida por alunos. A frequência dos alunos não está diretamente ligada à disciplina ou ao professor e sim com a aula a que o aluno assistiu num determinado dia e horário com um determinado professor de uma determinada disciplina. Como não podemos relacionar o aluno com o relacionamento, devemos criar uma entidade associativa “Aula”, que contém os atributos identificadores de disciplina e de professor, além das informações sobre a aula a que ele as- sistiu (data, horário e tema, por exemplo). A partir daí, podemos relacionar o aluno com a aula a que ele assistiu.

Veja:



Imagem 08: A imagem mostra a entidade disciplina relacionada com professor e a conexão representada pelo losango ministra com outro losango com nome frequenta que se relaciona com aluno, a relação entre disciplina e professor é conhecida como entidade associativa

e tornou-se uma entidade.

O relacionamento “ministra” tornou-se uma entidade associativa “Aula”.

**Tipos de Relacionamentos:**

Após a identificação das entidades, deve-se definir a forma como elas se relacionam entre si. De acordo com a quantidade de objetos envolvidos em cada lado do relacionamento, podemos classificá-los de três formas:

**Relacionamento 1..1 (um para um):**cada uma das duas entidades envolvidas referenciam obrigatoriamente a

uma única unidade da outra.

Porexemplo, em um banco de dados de currículos, cada usuário cadastrado deve possuir necessariamente apenas um currículo na base, ao mesmo tempo em que cada currículo só pertence a um único usuário cadastrado.



Imagem 09: A imagem mostra que a cardinalidade mínima e máxima é 1,1 , ou seja, um usuário pode ter no mínimo 1 e no máximo 1 currículo. Da mesma forma, um currículo pertence a no mínimo 1 e no máximo a 1 pessoa.

Nesse caso, dizemos que a cardinalidade mínima e máxima é 1,1 , ou seja, um usuário pode ter no mínimo 1 e no máximo 1 currículo. Da mesma forma, um currículo pertence a no mínimo 1 e no máximo a 1 pessoa.

**Relacionamento1..nou1..\*(umparamuitos):**uma das entidades envolvidas pode referenciar a várias unidades da outra, porém, do outro lado cada uma das várias unidades referenciadas só pode estar ligada uma unidade da outraentidade.

Por exemplo, em um sistema de biblioteca, uma editora pode publicar vários livros, mas cada livro pode ser publicado por apenas uma editora.

**(1,1)**

**(1,N)**

**Editora**

**publica**

**Livros**

Imagem 10: A imagem mostra que a cardinalidade mínima e máxima de Editora é 1,N , ou seja, uma editora pode publicar no mínimo 1 e no máximo muitos (indeterminados) livros.

Já um livro pode ser publicado por no mínimo 1 e no máximo 1 editora.

Nesse caso, dizemos que a cardinalidade mínima e máxima de Editora é 1,N , ou seja, uma editora pode publi- car no mínimo 1 e no máximo muitos (indeterminados) livros.

Já um livro pode ser publicado por no mínimo 1 e no máximo 1 editora.

**Relacionamenton..nou\*..\*(muitosparamuitos):** neste tipo de relacionamento cada entidade, de ambos os lados, podem fazer referência a múltiplas unidades daoutra.

Por exemplo, em um sistema de controle de projetos, um engenheiro pode desenvolver vários projetos e ao mesmo tempo um projeto pode ser desenvolvido por vários engenheiros. Assim, um objeto do tipo engenheiro pode referenciar múltiplos objetos do tipo projeto, e vice versa.

**(1,N)**

**(0,N)**

**Engenheiro**

**desenvolve**

**Projeto**

Imagem 11: A imagem mostra que a cardinalidade mínima e máxima de Engenheiro é 0,N , ou seja, um engenheiro pode desenvolver no mínimo 0 (nenhum) e no máximo muitos (indeterminados) projetos.

Esses projetos podem ser desenvolvidos por no mínimo 1 e no máximo muitos (indeterminados) engenheiros.

Nesse caso, dizemos que a cardinalidade mínima e máxima de Engenheiro é 0,N , ou seja, um engenheiro pode desenvolver no mínimo 0 (nenhum) e no máximo muitos (indeterminados) projetos.

Esses projetos podem ser desenvolvidos por no mínimo 1 e no máximo muitos (indeterminados) engenheiros.

Adaptado de https:[//www.devmedia.com.br/modelo-entidade-relacionamento-](http://www.devmedia.com.br/modelo-entidade-relacionamento-)

**Tipos de Atributos**

Os atributos podem ser classificados da seguinte forma:

**AtributoComposto:**Umatributoformadoporváriosatributosquenãopodeserdividid o,échamadode atômico ousimples.

Exemplo: O atributo endereço pode ser decomposto por vários valores que o compõem. Um endereço é com-

posto por nome da rua, número do prédio, cidade, estado e CEP. Observe a sua

rua

Residência

número

Estado

cidade

CEP

representação:

Imagem 12: A imagem mostra a elipse residência representada por várias elispes com os atributos: rua, número, cidade, cep e estado conhecido como atributo composto.

**Atributo monovalorado:**É um atributo que assume um único valor para uma determinadaentidade.

Exemplo: O nome de uma pessoa é um atributo monovalorado por guardar um único valor para cada pessoa. Repare na sua representação:

Nome

Imagem 13: A imagem mostra o atributo monovalorado representado por uma eilpse com o campo nome.

**Atributomultivalorado:** É um atributo que pode assumir diversos valores para uma determinada entidade.

Exemplo: o atributo telefone pode guardar vários valores para cada registro. Provavelmente você tem pelo menos dois números de telefones diferentes para contato: o número do seu celular, o telefone da sua casa ou do trabalho, então, o telefone pode ser considerado um atributo multivalorado por guardar mais de um valor para uma mesma pessoa. Repare que o atributo composto é representado por uma linha dupla em

Telefone

Imagem 14: A imagem mostra o atributo multivalorado representado por uma eilpse de linha dupla com o campo telefone.

**AtributoDerivado:**Um atributo pode ser chamado de derivado quando seu valor é determinado a partirde um ou mais atributos. Estes atributos mantêm uma “relação”. Em alguns casos, dois ou mais atributos estão relacionados.

Porexemplo, a idade e data de nascimento de um cliente. Para um determinado cliente, podemos determinar a sua idade por meio da data de nascimento e da data atual. Atributos como a idade são chamados de atributos derivados. Repare que o atributo derivado é representado por uma linha pontilhada em seu contorno:

Telefone

Imagem 15: A imagem mostra o atributo derivado representado por uma eilpse pontilhada com o campo telefone.

Uma chave primária é um atributo que possui um valor único para cada entidade individual. Em outras pala- vras, é quem garante a unicidade de uma entidade e nunca poderá ser nula. Esta é uma restrição que proíbe que duas entidades possuam o mesmo valor para um determinado atributo ao mesmo tempo. Alguns tipos de entidades podem ter mais que um atributo formando uma chave ou mais que um atributo chave.



**Chave Primária**

Uma chave primária é um atributo ou conjunto de atributos que identifica unicamente um registro na tabela e que por isso não pode ser repetido.

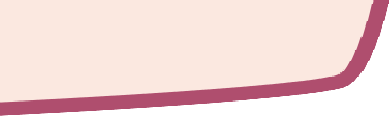
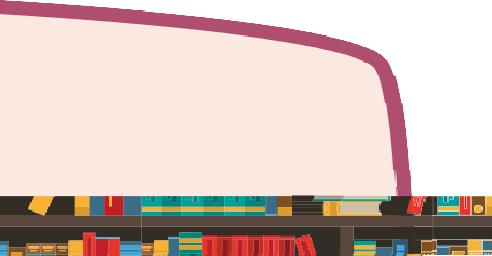
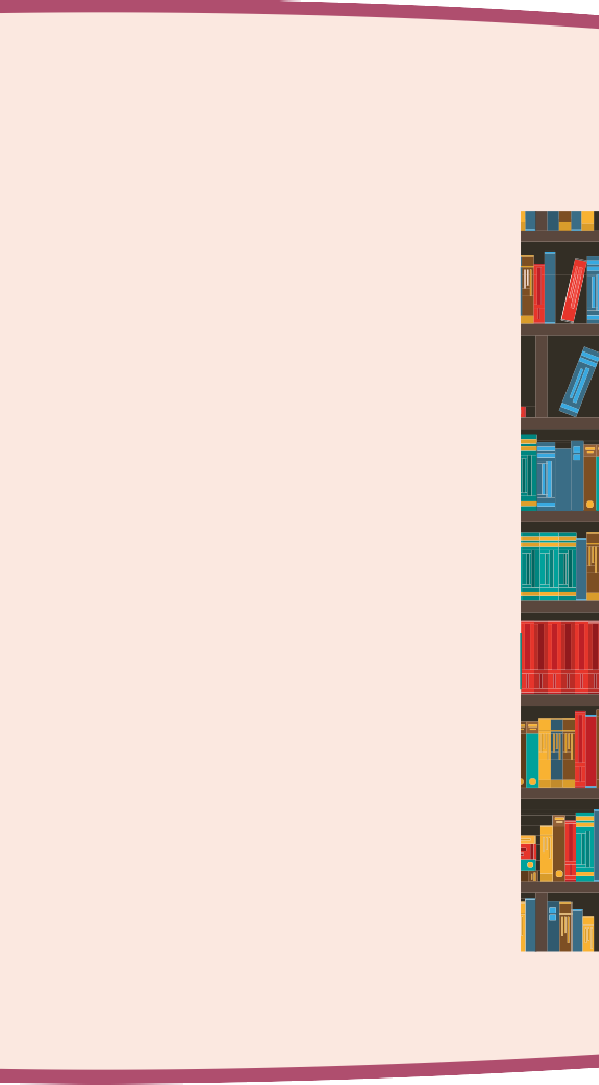
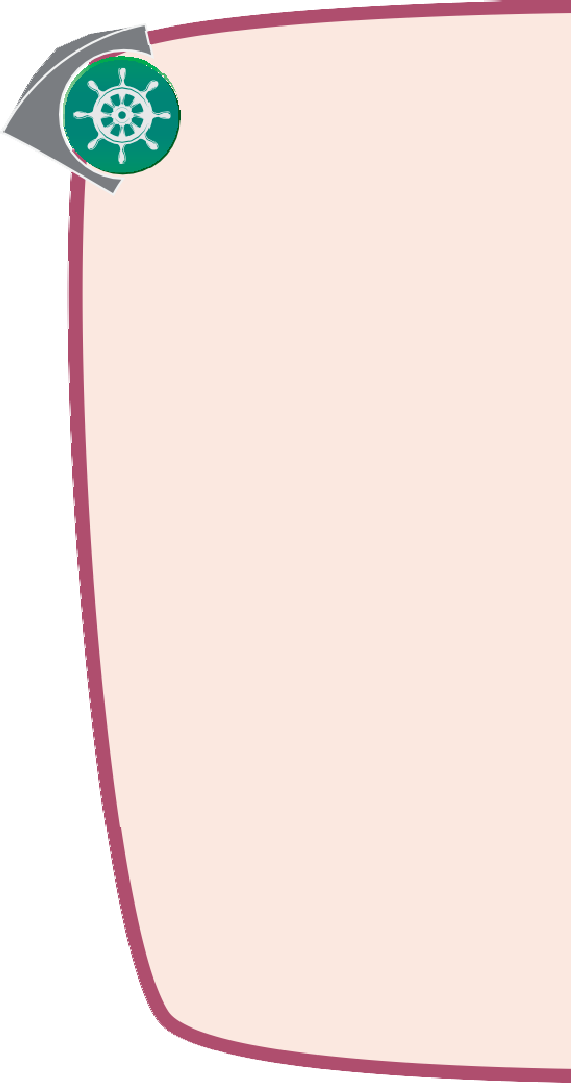
Por exemplo: em um cadastro de clientes, podemos ter as informações do número do CPF, nome, telefone, renda mensal etc. No caso de uma consulta ao Banco de Dados, precisamos de um campo que identifique o cliente. O campo CPF é único para cada cidadão, portanto, se fizermos uma pesquisa por esse dado, com certeza, a consulta retornará apenas o registro do cliente daquele CPF. O atributo chave primária é representado pelo nome do campo grifado. Veja:



Imagem 16: A imagem mostra e entidade cliente composta pelos atributos nome, renda mensal, telefone e a chave primária CPF grifada.

**Com base nos conceitos modelagem ER, como vo abaixo?**

**q**



**ue estudamos sobre**

**ê esquematizaria a situação**

**c**

**Uma biblioteca deseja informatizar os Cadastros de seus livros. Para isso a bibliotecária nos passou as seguintes informações:**

**- Os livros são definidos pelo número do ISBN, título, número de páginas e preço. Um livro pode ser escrito por vários autores, sendo os autores definidos pelo código, nome e formação. As editoras são definidas pelo código da editora, nome, endereço e e-mail.**

**Alguns livros podem ter várias cópias cadastradas na**

**biblioteca, sendo estas c pias definidas pelo número**

**ó**

**da cópia e data de aquisição desta cópia. É claro, que só podemos ter a cópia de algum livro na biblioteca se existir o livro , caso contrário, não é permitido.**

**Uma editora pode publicar vários livros, mas um livro**

**pode ser publicado por uma única editora.**

Imagem 17: Representa uma estante repleta de livros, todos coloridos. Pixabay.

Para desenhar um Diagrama ER, primeiramente precisamos nos atentar aos requisitos do sistema, ou seja, perceber quais são as necessidades do cliente e entender seu funcionamento.

Com base nas informações da bibliotecária e nos conceitos de modelagem que estamos estudando nessa agenda, faça o levantamento de todas as entidades e realize os relacionamentos entre elas e definir os seus atributos.

Em seguida, confira se você acertou!

1- Identificar as entidades existentes no sistema,temos:

Para saber se um determinado conceito é uma entidade, você deve sempre se perguntar se deseja armazenar informações sobre esse conceito e se há informações a serem armazenadas. Se tiver, você pode identificá-lo como uma entidade.

livro

autor

editora

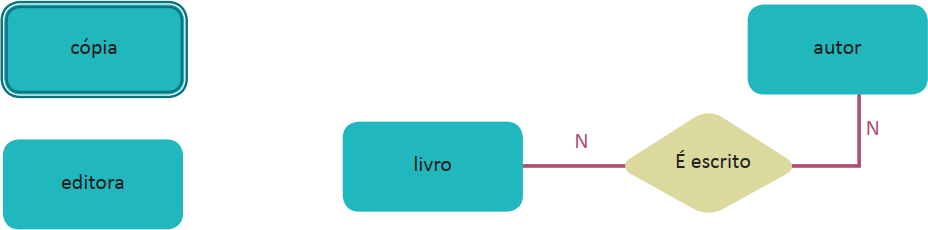
cópia

Com base na informação:” ... só podemos ter a cópia de algum livro na biblioteca se existir o livro”, definiremos a entidade cópia como entidade fraca, portanto, representada por uma caixa dupla.

2- Identificarosrelacionamentosentreasentidades,definindosuascardinalidades,dianted asinformaçõesque temos sobre o funcionamento dosistema:

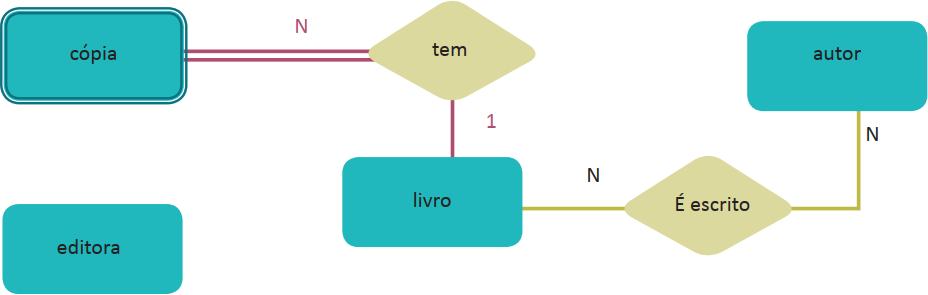
Analise o escopo do problema:

- Com base na informação “Um livro pode ser escrito por vários autores” e, é claro, que um autor pode escrever vários livros, podemos definir o seguinte relacionamento:



- Com base na informação “Alguns livros podem ter várias cópias cadastradas na biblioteca...”, podemos definir o

seguinte relacionamento:

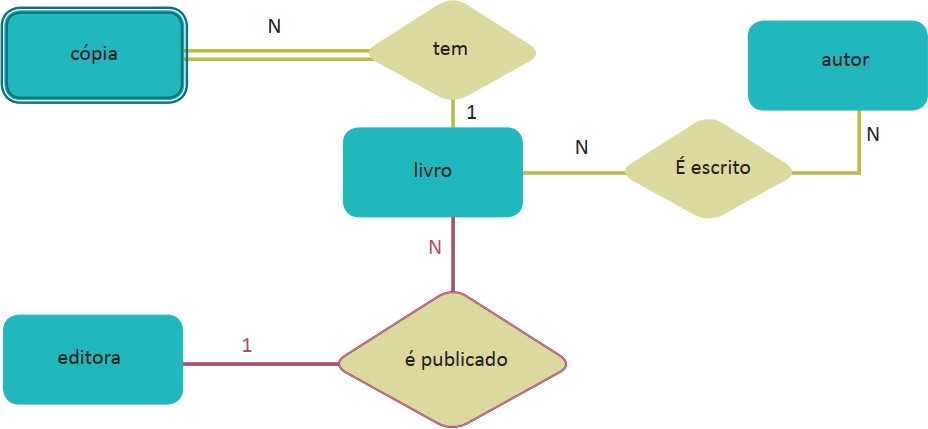


Observe que pelo fato da cópia ser uma entidade fraca, o relacionamento é representado por um traço duplo

também.

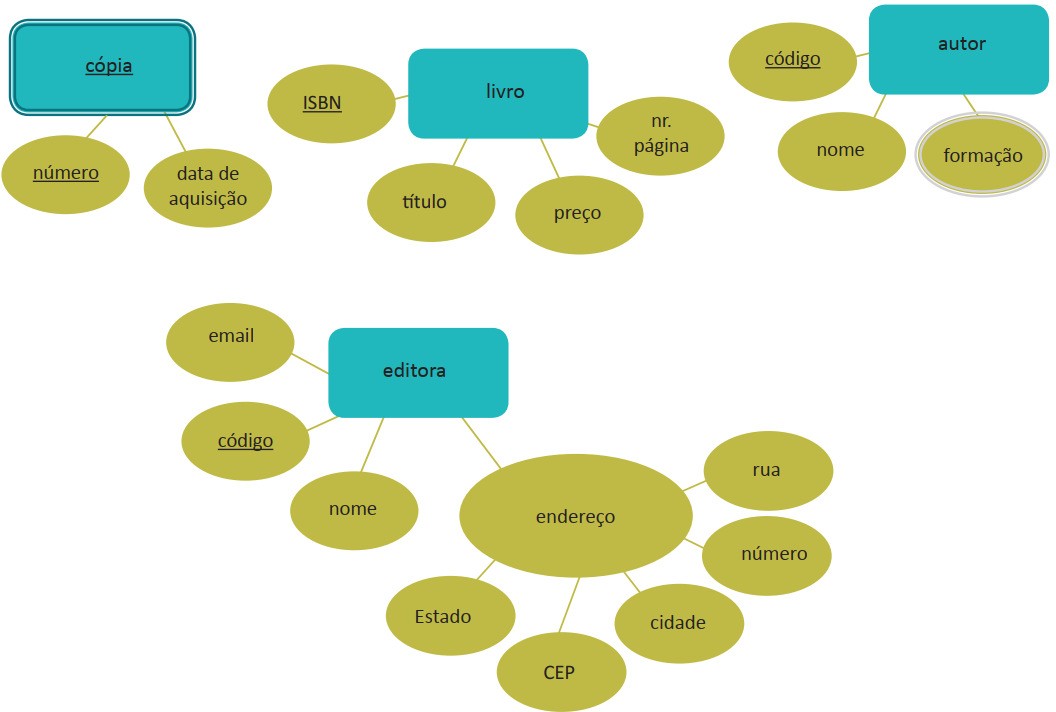
- Com base na informação de que: “Uma editora pode publicar vários livros, mas um livro pode ser publicado por

uma única editora”, criaremos o seguinte relacionamento.



3- Identificar quais são os atributos de cada entidade e a representação dos seustipos.

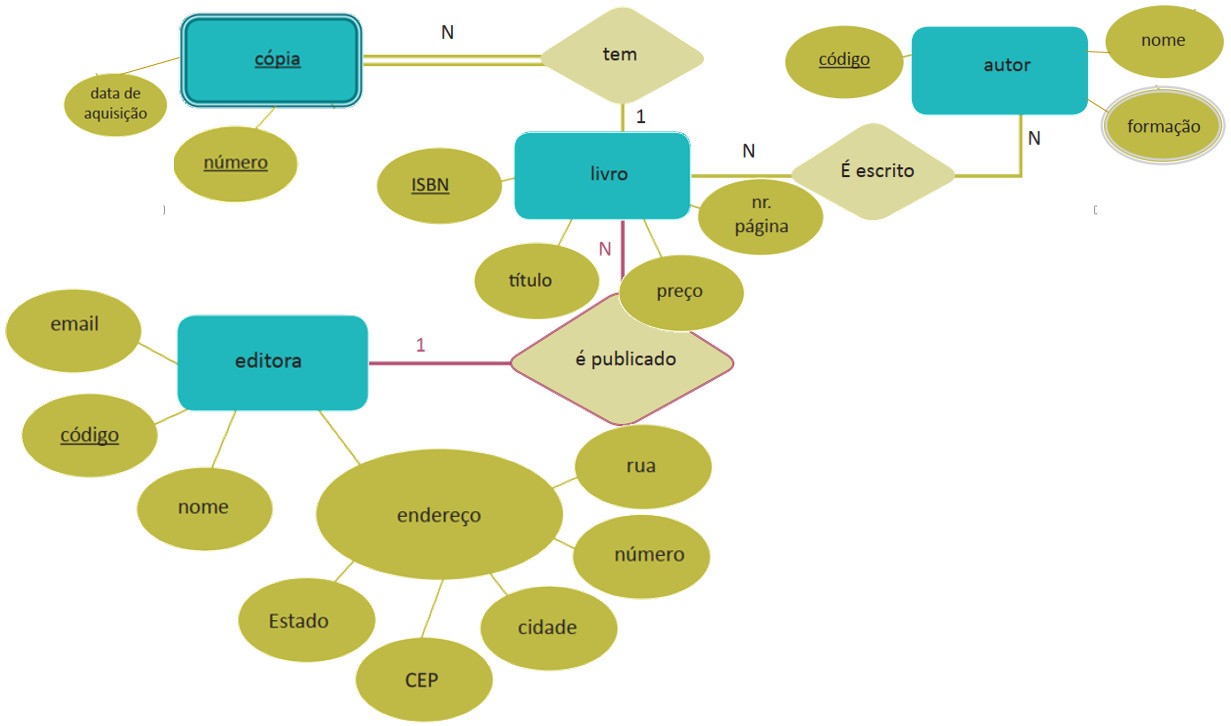
Lembrando que os atributos qualificam e descrevem a entidade.



Observe que o atributo “Formação” do autor é multivalorado. Você sabe por quê?

Bem, uma pessoa pode ter mais do que uma formação, não pode? Um professor pode ser matemático, filósofo e pedagogo. Por isso, definiremos o atributo formação do tipo multivalorado para poder armazenar mais de um valor para esse campo. Entendeu?

Assim, o Diagrama Entidade Relacionamento deverá ficar desta forma:



Agora, assista ao vídeo sobre Modelo ER com o Prof. Sandro Valérius, disponível em:

**https:**[**//www.youtube.com/watch?v=dMOLZ7PBwgY.**](http://www.youtube.com/watch?v=dMOLZ7PBwgY) Acessado em 20/09/2018.



Monte o Diagrama Entidade Relacionamento (DER), a partir dos enunciados a seguir:



1. Umaempresadesejamantercontroledosprojetosdesenvolvidosporseusdepart amentos.Temos as seguintesinformações:

Cadadepartamentoéresponsávelporumúnicoprojetoemdesenvolvimento.Estesproje tosdesenvolvemequipamentospara motoreshidráulicos.Umprojetoenvolveváriaspeçaseestaspeçasestãoenvolvidasemvári osprojetos. Em um departamento trabalham vários funcionários, mas para cada departamento há um único gerente responsável.

Departamentos: Número do departamento, Nome e Local. Projetos: Código do Projeto, Descrição e Duração.

Peças: Número da Peça e Descrição.

Gerente e Funcionários: Matrícula, Nome e Cargo.

1. Deseja- secriarumbancodedadosparaumaagênciadeturismo,contendoasinformaçõessobreosre cur- sos oferecidos por algumascidades:

Cada cidade possui hotéis, restaurantes e casas de show. Deseja-se armazenar os dados dos shows típicos de cada casa de show e os cardápios principais de cada restaurante da cidade. Para cada hotel existe uma pessoa responsável com quem a agência de turismo deve manter contato (para cada hotel existe uma única pessoa que cuida da hospedagem desta agência de turismo e cada pessoa é responsável por um único hotel). Os cardápios somente devem ser cadastrados se forem realmente utilizados pelos restaurantes.

Cidade: Cep, nome, estado.

Hotel: Cnpj, nome, endereço(rua, numero, bairro)

Casa de Show: Cnpj, dias de funcionamento, horário, nome, telefone. Restaurante: Cnpj, nome, telefone, endereço.

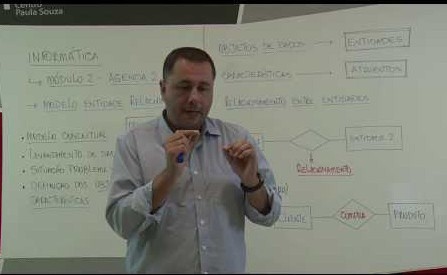
Cardápio: Código, nome, descrição(café da manhã, almoço e jantar). Responsável pela agencia de turismo: CPF, nome, telefone.

Salve os Diagramas e envie para o seu professor tutor.

AMPLIANDO HORIZONTES



Não deixe também de assistir aos vídeos:

Modelo Entidade-Relacionamento. Esse vídeo explica os primeiros conceitos do Modelo E-R, bem como a sua utilização. Disponível em

**https:**[**//www.youtube.com/watch?v=6AnCIwyoz-o**.](http://www.youtube.com/watch?v=6AnCIwyoz-o) Acessado em 17/09/2018.

Curso MySQL#14 Modelo Relacional. Esse vídeo, apesar de um pouco longo, traz explicações com ilus- trações interessantes sobre relacionamentos entre entidades. Disponível em **https:**[**//www.youtube.com/watch?v=8fxKJWJcRTw**.](http://www.youtube.com/watch?v=8fxKJWJcRTw) Acessado em 17/09/2018.



Para aprofundamento dos temas discutidos nesta agenda, seguem algumas dicas de livros que se relacionam com o conteúdo estudado. Estas dicas são muito importantes para você!

**Livros:**

ELMASRI, R.; NAVATHE, S. B. Sistemas de bancos de dados. 6 ed. Pearson, 2012. TAKAHASHI, M.; AZUMA, S. Guia mangá de banco de dados. 2 ed. Novatec: 2011. DATE, C.J. Introdução a Sistema de Banco de Dados. 8 ed. Campus, 2013



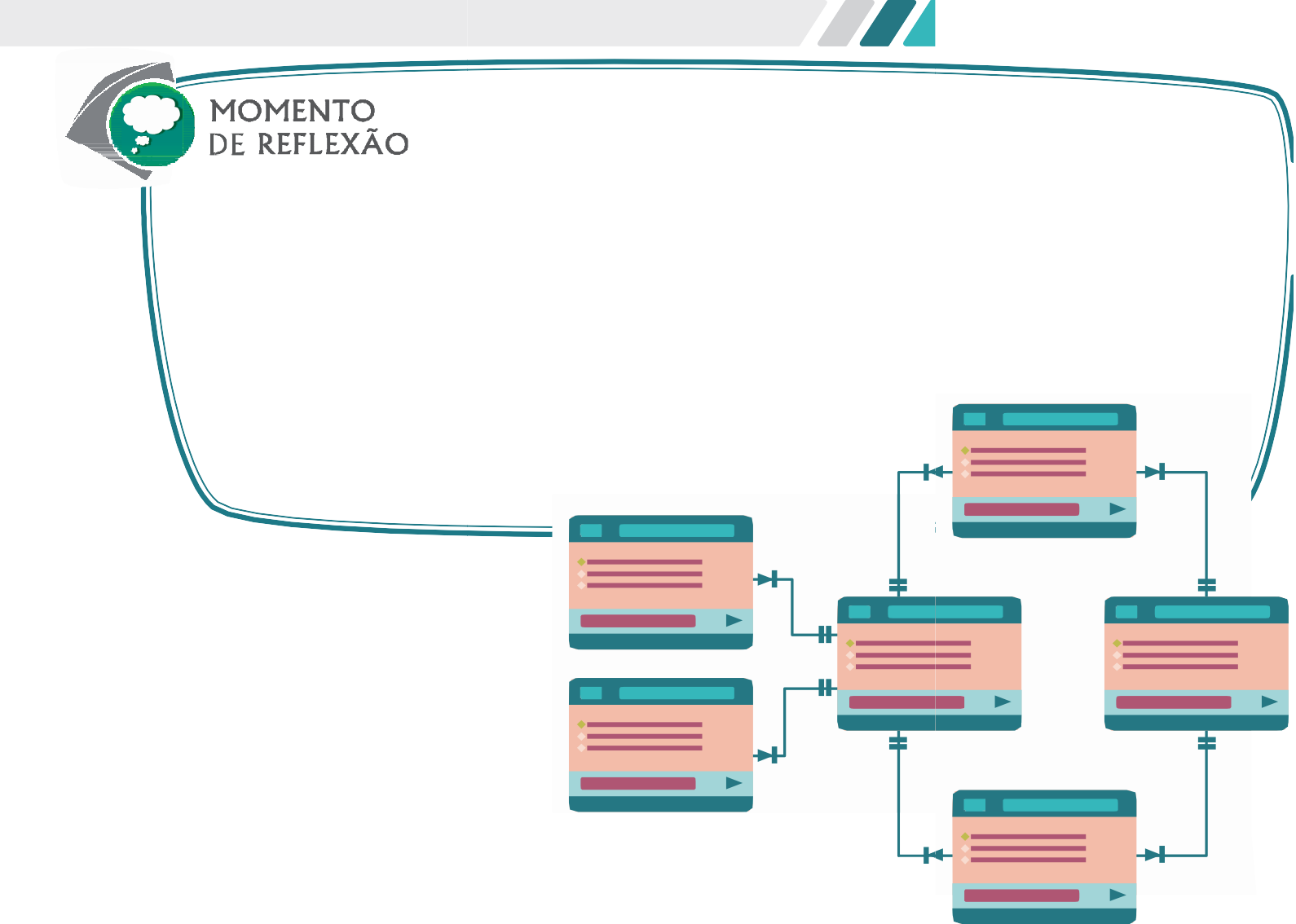
## AGENDA 11

### MODELO LÓGICO E MODELO FÍSICO: INTRODUÇÃO

**À LINGUAGEM**

**SQL ANSI**

***MODELO LÓGICO E SQL***



***Imagine que você deseja organizar os dados pessoais dos seus amigos como: nome, endereço e telefone, utilizando um editor de textos. Qual recurso usaria? Provavelmente, uma tabela. Isso mesmo!***

***Nesta agenda, você irá aprender a organizar os dados coletados em tabelas; mas, agora, utilizando conceitos de Banco de Dados. Vamos lá?***

Antes de selecionar o Sistema Gerenciador de Banco de Dados (SGBD), é necessário fazer uma descrição dos tipos que informações que serão armazenadas em um banco de dados.

Para organizar essas informações utilizamos modelos, em diferentes níveis de abstração e com diferentes objetivos. Cada descrição recebe o nome de **Esquema de Banco de Dados**.

Imagem 01: A imagem mostra várias tabelas relacionadas entre si respresentando um Banco de Dados.

Após selecionar o SGBD, o projetista deverá relacionar as características e restrições do modelo conceitual com as do modelo selecionado para implementação. O modelo lógico constitui uma representação específica de um modelo interno, utilizando as estruturas de BD suportadas pelo banco escolhido. Porém, permite registrar **quais** dados podem aparecer no banco, mas não registra **como** estes dados serão armazenados no SGBD. Em um Banco de Dados Relacional (BDR), o esquema interno é expresso utilizando linguagem SQL, por padrão.

Neste nível, o modelo lógico depende do software. Portanto, qualquer alteração feita no SGBD exige que o modelo interno seja alterado para adequar-se às características e exigências de implementação do modelo de BD.



*33*



***MODELO LÓGICO E SQL***



A



Adriano é um rapaz empreendedor e está abrindo um minimercado em seu bairro. Já está quase tudo pronto, mas agora precisou contratar o Ferrari, um desenvolvedor de sistemas que ficará responsável pela implantação de um software que fará todo o controle de clientes, vendas, produtos em estoque e demais dados para controlar e organizar todas as informações do seu minimercado.

Ferrari fez o modelo conceitual por meio do levantamento dos dados do minimercado, classificando e organizando todos os requisitos levantados e depois partiu para o modelo lógico, agrupando os dados em entidades e relacionamentos por meio do DER (Diagrama Entidade e Relacionamento).

Depois, Ferrari explicou para Adriano que precisavam definir um Sistema Gerenciador de Banco de Dados (SGBD) para implementar as informações coletadas. Aproveitou a ocasião e explicou ao proprietário do estabelecimento que existem inúmeras opções de SGBD no mercado que implicam vantagens e desvantagens. Assim, diante da necessidade do minimercado, escolheram o Banco de Dados MySQL e, a partir das especificações deste SGBD, Ferrari construiu o DER, apresentado a seguir, fazendo os possíveis refinamentos para alcançar maior desempenho das operações sobre o BD escolhido.

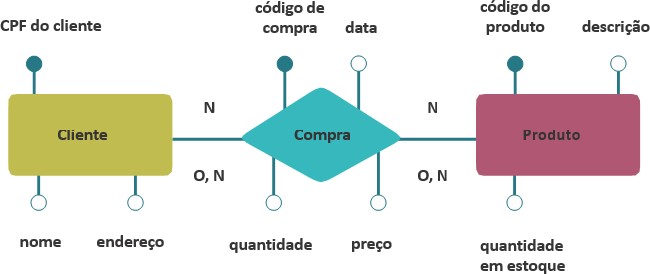


Imagem 02: A imagem mostra a entidade cliente relacionada com a entidade produto. Cliente tem os atributos: CPF (chave primária), nome e endereço e a entidade produto com os campos código do produto (chave primária), descrição e quantidade no estoque. Na relação entre as entidades um cliente pode adquirir vários produtos e um produto pode ser adquirido por vários clientes, sendo assim Compra tem como atributos: código da compra (chave primária), data, preço e quantidade.

Ferrari precisa garantir que as informações contidas no Projeto Lógico sejam representadas de acordo com

o modelo relacional. Para isso, seguiu algumas diretrizes com objetivo de restringir as opções dos desenvolvedores em um conjunto efetivo de regras que estudaremos nessa agenda.

*34*

 ***MODELO LÓGICO E SQL***

Antes de continuar a construção do Diagrama ER para o minimercado do Adriano, vamos entender quais os elementos necessários para montar o Modelo Lógico.

O Modelo Lógico parte para a construção de um modelo em um formato adequado ao Sistema Gerenciador de Banco de Dados onde o projeto será implementado. Sendo assim, os modelos são bastante dependentes da tecnologia a ser adotada (relacional, redes, orientado a objetos).

Para o processo de obtenção de um modelo lógico, a partir de um modelo conceitual, é preciso aplicar deter- minadas regras de derivação. Como nosso objetivo é explorar a derivação de modelos relacionais, trabalharemos com as regras para o modelo relacional, segundo Elmasri e Navathe (2011).

No modelo lógico, além de definir as entidades, atributos e relacionamentos, temos também que identificar as chaves primárias, chaves estrangeiras e tratar os relacionamentos entre as entidades.

d



**Relembrando o conceito de Chave Primária:**

Uma chave primária é um atributo que possui um valor único para cada entidade individual. Em outras

palavras, **Chaves Primárias** (**Primary Keys** ou “**PK”**), sob o ponto de vista de um banco de dados são representadas por palavras que garantem a unicida e de uma entidade e nunca poderá ser nula. Esta é uma restrição que proíbe que duas entidades possuam o mesmo valor para um determinad atributo ao mesmo tempo, ou seja, elas nunca se repetem.

Alguns tipos de entidades podem ter mais que um atributo formando uma chave ou mais que um atributo chave.

Por exemplo, pense num cadastro que você faz numa loja, solicitando o seu CPF, seu nome, telefone e endereço.

o

Quais desses dados (atributos) podem garantir que durante uma consulta no banco de dados da loja, podem trazer somente o seu cadastro?

**Nome?**

Bom, vamos pensar e se você se chama João de Souza ou Maria da Silva? Pode acontecer de haver mais de

um cliente com o mesmo nome, por ser mais comum, não pode?

Então, nome não é um atributo que garante que a consulta retorne um único registro, portanto, não poderá ser escolhido como chave primária.

Endereço?

Hummm, você e seu irmão podem ser clientes da loja e, portanto, o endereço pode ser o mesmo, não é? Então, o endereço também não é um atributo que garante que a consulta retorne um único registro, portanto, da mesma forma não poderá ser escolhido como chave primária.



*35*

Telefone?

O mesmo problema do endereço! Você e o seu irmão poderão informar o mesmo telefone no cadastro. Então, telefone não é legal para ser escolhido como chave primária.

E o CPF?

Já o Cadastro de Pessoa Física (CPF) é único para cada cidadão brasileiro, portanto, CPF pode ser escolhido como chave primária.

Entendeu o conceito de chave primária?

A simbologia utilizada para representar uma chave primária é o sublinhado no nome do campo ou uma bolinha fechada. Veja:

CPF



pessoa

CPF

pessoa

ou

Imagem 03: A imagem demonstra as formas para representar uma chave primária. A entidade pessoa tem como chave primária a palavra CPF grifada dentro de uma elipse. A outra imagem tem a entidade pessoa por uma linha com um círculo preenchido.

E o conceito de **Chave Estrangeira**. Você sabe o que é?

Quando uma entidade apresenta um atributo importado de outra entidade, esse atributo é denominado

**chave estrangeira**. As chaves estrangeiras são o resultado de associações entre entidades.

Por exemplo, imagine que você queira cadastrar vários produtos que sejam de uma determinada categoria. Toda vez que você preencher os dados do produto, precisaremos indicar a chave primária da tabela categoria que seja da categoria que o nosso produto pertencerá. Ou seja, quando inserirmos um registro na tabela de produtos com o “id\_categoria”, essa chave primária da tabela “categorias” representará uma chave estrangeira (FK) dentro da tabela de produtos. É uma chave que vem de fora, de outra tabela, por isso o nome “estrangeira”.

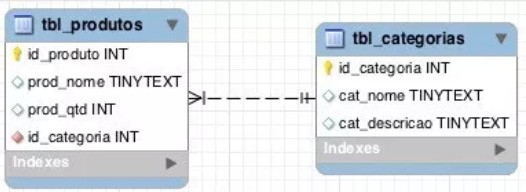
Com essa chave estrangeira, podemos facilitar as consultas e fazer cruzamento de dados por meio destas referências. Dessa forma, uma chave estrangeira sempre será necessária quando houver o relacionamento entre duas tabelas.

Imagem 04: A imagem demonstra o relacionamento entre uma tabela de produtos e uma tabela de categorias e o relacionamento entre elas por meio da chave estrangeira.

Adaptado de https://[www.diegomacedo.com.br/entendendo-as-chaves-dos-bancos-de-dados/.](http://www.diegomacedo.com.br/entendendo-as-chaves-dos-bancos-de-dados/) Acessado em 05/04/2019.

Só para descontrair...

Imagem 05: https://vidadeprogramador.com.br/2017/03/15/chave-estrangeira/. Acessado em 05/06/2019. Chefe: Jimmy, sabe a parte de produtos e categorias que você fez?

Jimmy: Sei

Chefe: Fiz um teste e excluí a categoria "parafusos". O sistema apagou sem dar nenhuma mensagem, agora tem um monte de produtos perdidos, sem categoria.

Jimmy: Ah, é que eu não uso esse negócio de chave estrangeira que os piás aí usam... Vou arrumar. Programador: PLOFT!

Veremos exemplos de chave estrangeira nas regras de mapeamento. Vamos lá!

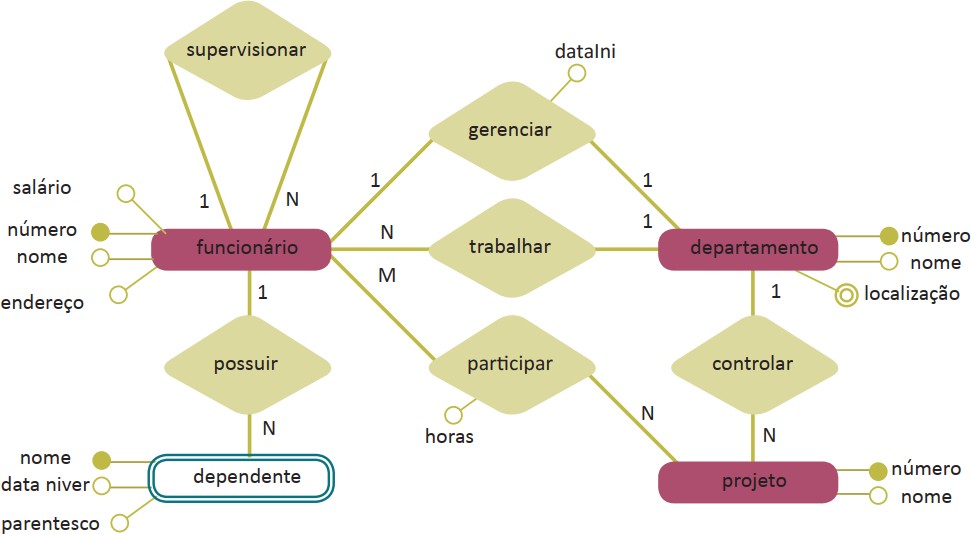
**Regras de Derivação – Mapeamento**

Uma vez que se consegue representar a semântica de uma aplicação por meio de modelos de dados, em especial pelo Modelo Entidade Relacionamento, é importante garantir que as informações contidas neste modelo sejam representadas corretamente no Modelo Relacional. Para garantir uma representação fiel das in- formações no Modelo Relacional é preciso seguir algumas diretrizes, com objetivo de restringir as opções dos desenvolvedores em um conjunto efetivo de regras.

Vamos às regras do mapeamento:

1. Mapear Conjuntos de Entidades Regulares

Identificar todas as entidades regulares (não fracas) e seus atributos



**Funcionário =** {**f**número, **f**nome, **f**endereço, **f**salário} **Departamento =** {**d**número, **d**nome}

**Projeto =** {**p**número, **p**nome} Observações:

1. Por padronização, adotamos um sufixo no nome dos atributos para identificar a qual entidade esse atributo pertence. Por isso, os atributos da entidade Funcionário iniciam-se sempre com a letra “f”; da entidade Departamento com a letra “d”, da entidade Projeto iniciam com a letra “p” e assim por diante.
2. O atributo “localização” não foi representado na relação “Departamento” porque é um atributo multivalo- rado e será tratado na 6ª Regra.
3. O conjunto da entidade “Dependente”, por ser uma entidade fraca, será tratado na regra 2, a seguir.

2. Mapear Conjuntos de Entidades Fracas

Relacionar todos os atributos da entidade fraca, acrescentando também o atributo chave da entidade da qual ela depende.

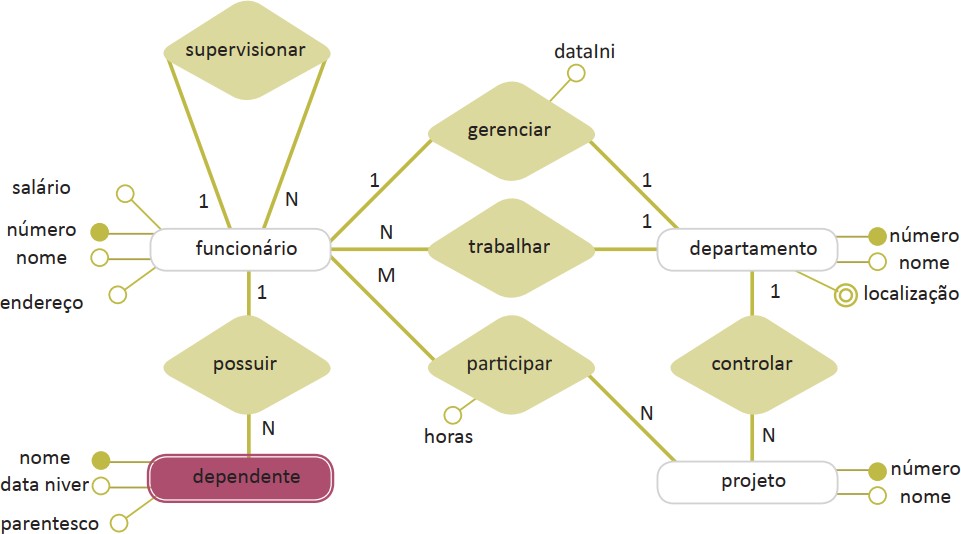


Imagem 07: A imagem demonstra um DER que representa as entidades, seus atributos e relacionamentos. Tem-se como entidades: funcionário, departamento, projeto e dependente. Funcionário está relacionado com as entidades: dependente, departamento e projeto. Na cardinalidade a entidade funcionário supervisiona muitos funcionários, e este funcionário possui vários dependentes, um funcionário gerencia um departamento, e este funcionário participa de vários projetos e os projetos tem muitos funcionários envolvidos. Um departamento pode controlar vários projetos.

**Dependente =** {**depend**nome, **f**número, **depend**dataniver**, depend**parentesco}

Observe que o atributo da entidade Funcionário (fnumero), vai para a entidade Dependentes como um atributo chave! Portanto, a entidade Dependente terá uma chave composta pelos atributos **nome do dependente e número do funcionário** do qual depende.

3. Mapear Conjuntos de Relacionamentos Binários 1:1

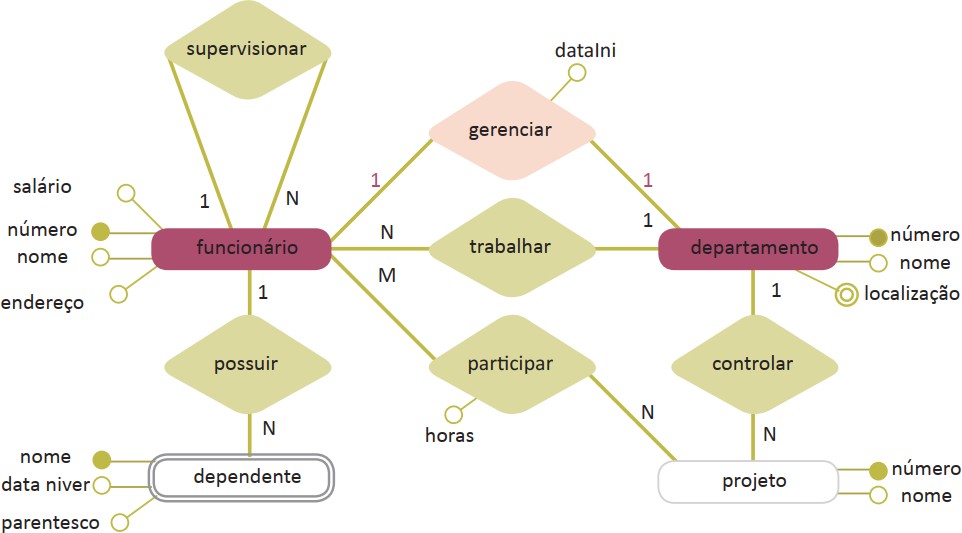
No mapeamento do relacionamento 1:1, deve-se escolher uma entidade para receber o atributo chave da outra entidade relacionada.

Imagem 08: A imagem demonstra um DER que representa as entidades, seus atributos e relacionamentos. Tem-se como entidades: funcionário, departamento, projeto e dependente. Funcionário está relacionado com as entidades: dependente, departamento e projeto. Na cardinalidade a entidade funcionário supervisiona muitos funcionários, e este funcionário possui vários dependentes, um funcionário gerencia um departamento, e este funcionário participa de vários projetos e os projetos tem muitos funcionários envolvidos. Um departamento pode controlar vários

Para escolher qual entidade deverá receber o atributo da outra entidade, devemos sempre fazer uma pergunta envolvendo essas duas entidades relacionadas. Nesse caso, devemos perguntar:

Todo funcionário gerencia um departamento? Resposta: - Não necessariamente!

Sabemos que nem todo o funcionário é gerente de um departamento, não é? Então, não é interessante que a entidade Funcionário receba a chave primária do departamento gerenciado.

Agora fazemos a pergunta ao contrário:

- Todo o departamento é gerenciado por um funcionário?

**Resposta: - Sim, todo o departamento deve ter alguém responsável por ele.** Concorda?

Sendo assim, departamento receberá a chave primária do funcionário gerente.

É importante observar também que existe um atributo no relacionamento (**dataini** – que é a data que o

funcionário começou a gerenciar o departamento. Esse atributo está representado no relacionamento porque não pertence apenas à entidade funcionário e nem somente à entidade departamento; mas, sim, ao relacionamento, pois existirá na relação funcionário gerencia departamento, indicando quando isso começou a acontecer.

Neste caso, o mapeamento do relacionamento entre essas duas entidades ficará assim:

**Departamento =** {**d**número, **d**nome, **f**numerogerente, **gerencia**dataini}

O atributo recebido pela outra entidade é chamado de Chave Estrangeira. Portanto, fnumerogerente do empregado é a chave estrangeira da entidade Departamento.

Note que o atributo chave da uma entidade Funcionário vai para a entidade Departamento como um atribu- to simples!

4. Mapear Conjuntos de Relacionamento Binário Regular 1:N

Na 4ª regra, deve-se acrescentar à entidade de lado N, o atributo chave da entidade de lado 1.

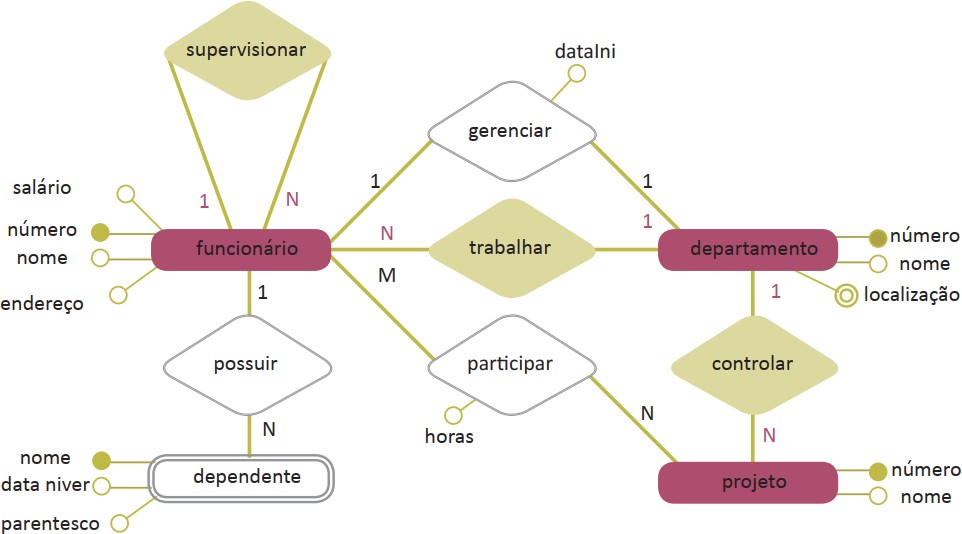


Imagem 09: A imagem demonstra um DER que representa as entidades, seus atributos e relacionamentos. Tem-se como entidades: funcionário, departamento, projeto e dependente. Funcionário está relacionado com as entidades: dependente, departamento e projeto. Na cardinalidade a entidade funcionário supervisiona muitos funcionários, e este funcionário possui vários dependentes, um funcionário gerencia um departamento, e este funcionário participa de vários projetos e os projetos tem muitos funcionários envolvidos. Um departamento pode controlar vários projetos.

Neste passo, são mapeados os relacionamentos: **supervisionar**, **trabalhar** e **controlar**.

- Avaliando o relacionamento “supervisionar”:

Esse relacionamento é **recursivo**, ou seja, nesse caso, na classe funcionário existe um único funcionário que é supervisor dos demais funcionários. Assim temos uma relação 1:N.

**Observação**: os relacionamentos recursivos, também chamados de auto-relacionamentos, são casos especiais, onde uma entidade se relaciona com si própria. Apesar de serem relacionamentos muito raros, a sua utilização é muito importante em alguns casos.

- Um funcionário supervisiona vários funcionários e os funcionários são supervisionados por um único supervisor.

Sendo assim, o atributo SuperNúmero é adicionado na entidade Funcionário:

**Funcionário =** {**f**número, **f**nome, **f**endereço, **f**salário, **super**numero}

O mesmo acontece com o relacionamento “trabalhar”: A entidade de lado N da relação recebe a chave primária do atributo chave da entidade de lado 1. Sendo assim, o atributo DNumero deve ser adicionado à entidade Funcionário. Ficando representado dessa forma:

**Funcionário =** {**f**número, **f**nome, **f**endereço, **f**salário, **super**número, **d**numero}

Da mesma forma, no relacionamento “controlar”, das entidades Departamento e Projeto, o lado N da relação, no caso, a entidade Projeto, recebe a chave primária do lado 1 do relacionamento, no caso, da enti- dade Departamento:

Ficando representado dessa forma:

**Projeto =** {**p**número, **p**nome, **d**numero}

Note que os atributos chave das entidades do lado 1 foram para as entidades chave do lado N como atributos simples!

5. Mapear Relacionamento Binário M:N

O relacionamento binário envolve dois conjuntos de entidades. A maior parte dos conjuntos de relacionamentos modelados em um BD é do tipo binário. Para cada relacionamento binário M:N cria-se uma relação. Os atributos da relação são os atributos do conjunto de relacionamento, juntamente com os atributos chave das relações que mapeiam os conjuntos de entidades envolvidos. A chave da relação é a concatenação dos atributos chave das relações que mapeiam os conjuntos de entidades envolvidos.



Imagem 10: A imagem demonstra um DER que representa as entidades, seus atributos e relacionamentos. Tem-se como entidades: funcionário, departamento, projeto e dependente. Funcionário está relacionado com as entidades: dependente, departamento e projeto. Na cardinalidade a entidade funcionário supervisiona muitos funcionários, e este funcionário possui vários dependentes, um funcionário gerencia um departamento, e este funcionário participa de vários projetos e os projetos tem muitos funcionários envolvidos. Um departamento pode

Observe o relacionamento Funcionário “participa” de Projeto. Nesse caso, um funcionário pode participar de vários projetos e um projeto pode ter a participação de vários funcionários, portanto, dizemos que o relaciona- mento é N:M.

Assim, devemos criar uma nova entidade, a partir do relacionamento “Participar”, adotando como chave primária a **concatenação** da chave primária de Funcionário e da chave primária de Projeto e mais o(s) atributo(s) que existir(em) no relacionamento, criando-se uma nova entidade:

**Participar =** {**f**num, **p**num, horas}

Por que isso? Veja nesse exemplo que para você conseguir cadastrar um mesmo funcionário em vários proje- tos, deverá fazer um registro para cada um dos projetos dos quais o funcionário participa e cadastrar o número de horas a serem trabalhadas, concorda?

Por exemplo:

MARIA trabalha no projeto de desenvolvimento do sistema MAP com carga horária de 20 horas semanais. MARIA trabalha no projeto de desenvolvimento do sistema GIGA com carga horária de 10 horas semanais.

MARIA trabalha no projeto de desenvolvimento do sistema EMPLACAR com carga horária de 14 horas semanais.

Da mesma forma,

JOÃO trabalha no projeto de desenvolvimento do sistema MAP com carga horária de 20 horas.

JOÃO trabalha no projeto de desenvolvimento do sistema EMPLACAR com carga horária de 22 horas.

Perceba que você poderá repetir o número da matrícula da Maria para vários números de projetos diferentes e poderá repetir os números de projetos para vários funcionários diferentes. A única coisa que você não poderá fazer é cadastrar duas vezes o mesmo funcionário para o mesmo projeto, não é? Ora, e como você poderá fazer para que isso não aconteça? A única forma é **colocar os dois atributos como chave primária** (número da matrícula do funcionário e número do projeto).

6. Mapear Atributos Multivalorados

A 6ª regra identifica os atributos multivalorados. Os atributos multivalorados possuem um ou mais valores para o mesmo.

Para este atributo, deve-se criar uma nova entidade, inserindo o atributo multivalorado e o atributo chave da entidade a qual ele pertence. **Ambos os atributos devem ser chaves primárias, compondo uma chave concatenada.**

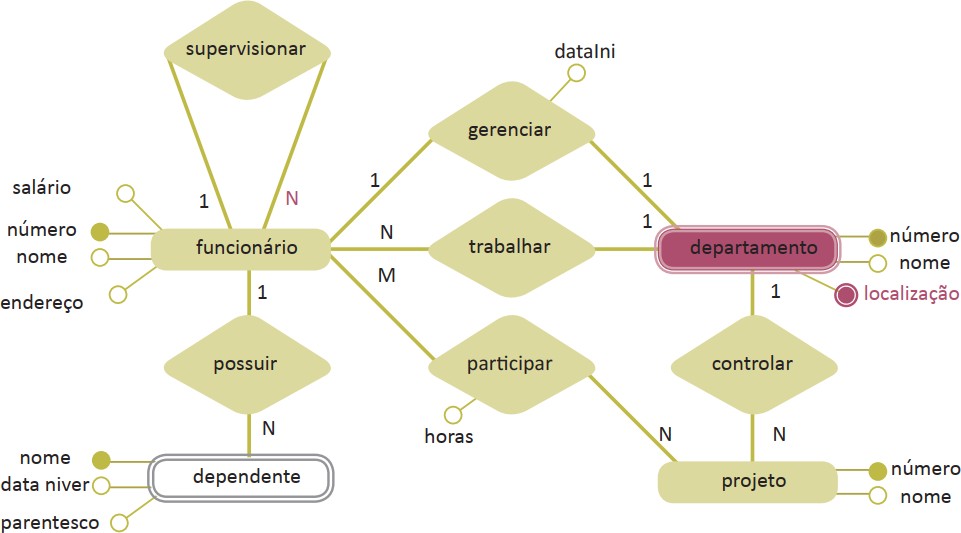


Imagem 11: A imagem demonstra um DER que representa as entidades, seus atributos e relacionamentos. Tem-se como entidades: funcionário, departamento, projeto e dependente. Funcionário está relacionado com as entidades: dependente, departamento e projeto. Na cardinalidade a entidade funcionário supervisiona muitos funcionários, e este funcionário possui vários dependentes, um funcionário gerencia um departamento, e este funcionário participa de vários projetos e os projetos tem muitos funcionários envolvidos. Um departamento pode controlar vários projetos.

**LocalDep =** {**d**número, localização} Entendeu o porquê?

Imagine que um departamento de uma universidade, por exemplo, esteja localizado em vários polos, em cidades diferentes. Para que não tenha o problema de cadastrar o mesmo polo, em uma mesma localização, mais de uma vez, definimos os dois atributos como chave.

Veja um exemplo:

**LocalDep:** (dnúmero, dlocalização)

Departamento número 10, local São Paulo. Departamento número 10, local Campinas. Departamento número 10, local Jundiaí.

Departamento número 20, local São Paulo. Departamento número 20, local Curitiba.

Perceba que apenas podemos dizer que um mesmo departamento tenha polos em localidades diferentes se o número do departamento e o local forem definidos como chave. Observe que dessa forma não é permitido cadastrar um mesmo departamento em um mesmo local mais de uma vez, evitando duplicidade de registros. Entendeu agora o porquê temos que definir o número do departamento e a sua localização como chave?

Sendo assim, o mapeamento do Modelo ER representado ficou dessa forma: Funcionário = {fnúmero, fnome, endereço, salário, supernúmero, dnumero}

Primeiro Passo: Quarto Passo:

Departamento = {dnúmero, dnome, fnúmero, dataini} Primeiro Passo: Terceiro Passo:

Projeto = {pnúmero, pnome, dnro} Primeiro Passo: Quarto Passo:

Dependente = {dependnome, fnúmero, dataniver, parentesco}

Segundo Passo:

Participar = {fnum, pnum, horas} Quinto Passo:

LocalDep = {dnúmero, dlocalização}

Sexto Passo:

VOCÊ NO COMANDO

Imagem 31: A imagem mostra três exemplos de DER para mapear os relacionamentos.



A imagem A tem a entidade empregado com a chave primária matrícula e os atributos nome e tel. relacionada com a entidade cartão de ponto com a chave primária número e o atributo mês, com cardinalidade de 1 para 1.

A imagem B tem a entidade empregado com a chave primária matrícula e os atributos nome e tel. relacionada com a entidade departamento com a chave primária código e o atributo nome, com cardinalidade de N para 1.

A imagem C tem a entidade aluno com a chave primária matrícula e os atributos nome e idade relacionada com a entidade disciplina com a chave

*46*

primária código e o atributo nome, com cardinalidade de N para M.

Mapeando os relacionamentos:

1. **Relacionamento 1:1, entre empregado e cartão de ponto: Um empregado possui um único cartão de ponto e um cartão de ponto pertence a apenas um empregado.**

Para escolher qual entidade deverá receber o atributo da outra entidade, devemos sempre fazer a seguinte pergunta: - Todo empregado possui um cartão de ponto?

Resposta: Sim! Por isso, a entidade Cartão de Ponto poderá receber a chave primária de Empregado

Agora fazemos a pergunta ao contrário: - Todo o cartão de ponto pertence a um empregado? Resposta: Sim, pois se existe o cartão de ponto é sinal que pertence a um empregado da empresa. Não é? Sendo assim, a entidade cartão de ponto também poderá receber a chave primária de empregado.

Então, nesses casos, você pode escolher qual entidade receberá a chave primária da outra. Vamos, então, levar o atributo chave da entidade Empregado para a entidade Cartão de Ponto:

Neste caso, o mapeamento do relacionamento entre essas duas entidades ficará assim:

Empregado – EmpMatrícula, EmpNome, EmpTelefone Cartão de Ponto – CpNúmero, CpMês, EmpMatrícula.

Dizemos que o atributo recebido pela outra entidade é chamado de **Chave Estrangeira**. Portanto, matrícula do empregado (**EmpMatrícula**) é a chave estrangeira de cartão de pontos.

1. Relacionamento 1:N (“um para N ou 1 para Muitos”): Um empregado pertence a um único departamen- to, mas um departamento possui muitos empregados.

Em um relacionamento 1:N, o lado N sempre recebe a chave primária do lado 1.

Para não esquecer, pense sempre nisso: muitos (N) são mais fortes do que 1 e por isso o lado N rouba a chave primária do lado 1! É uma brincadeirinha, mas ajuda a não errar!

Neste caso, o mapeamento do relacionamento entre essas duas entidades ficará assim:

Empregado – EmpMatrícula, EmpNome, EmpTelefone, DepCodigo Departamento – DepCodigo, DepNome

Dizemos que o atributo recebido pela outra entidade é chamado de **Chave Estrangeira**. Portanto, código do departamento (**DepCodigo** ) é a chave estrangeira da entidade empregado.

1. Relacionamento N:M (“N para M ou Muitos para Muitos”): Um aluno pode cursar mais de uma disciplina e uma disciplina pode ser cursada por mais de um aluno.

Os atributos da relação são os atributos do conjunto de relacionamento juntamente com os atributos chave das relações que mapeiam os conjuntos de entidades envolvidos. A chave da relação é a concatenação dos atributos chave das relações que mapeiam os conjuntos de entidades envolvidos. Por que isso?

Veja nesse exemplo que para conseguir cadastrar um mesmo aluno em várias disciplinas, deverá ser feito um registro para cada uma das disciplinas que o aluno cursa, concorda?

Exemplo:

Maria cursa matemática Maria cursa português Maria cursa história

E assim por diante Da mesma

forma:

João cursa matemática João cursa português João cursa história

Perceba que você poderá repetir a matrícula da Maria para vários códigos de disciplinas diferentes e poderá repetir os códigos das disciplinas para vários alunos diferentes. O que você não poderá fazer é cadastrar duas vezes o mesmo aluno para a mesma disciplina, não é?

Ora, e como você poderá fazer para que isso não aconteça?

A única forma é **colocar os dois atributos como chave primária** (matrícula do aluno e código da disciplina). Concorda?

Então, a terceira entidade criada pelo relacionamento N:M, terá como chave primária tanto a chave de Aluno quanto a chave de disciplina.

Veja como ficará o mapeamento:

Aluno – AlMatrícula, AlNome, AlIdade. Disciplina – DiscCodigo, DiscNome.

Curso – CurMatriculaAluno, CurCodDisciplina.

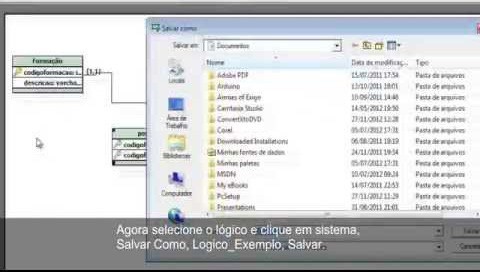
(junção) das chaves primárias de Aluno e Disciplina.

Assista ao vídeo do Prof. Sandro Valérius sobre Relacionamento entre tabelas.

Disponível em **https:**[**//www.youtube.com/watch?v=THZWI0jpkfI**.](http://www.youtube.com/watch?v=THZWI0jpkfI) Acessado em 21/09/2018.



Ferramenta Case BR Modelo para auxiliar no desenvolvimento do modelo lógico.

Para gerar o modelo lógico, existem algumas ferramentas case que auxiliam nesse trabalho. Uma delas é a ferramenta BR Modelo. Assista ao tutorial disponível em **https:**[**//www.youtube.com/watch?v=SP77QucKdlo**.](http://www.youtube.com/watch?v=SP77QucKdlo) Acessado em 20/07/2018:

Agora que você já conhece os conceitos do Modelo ER e como desenvolvê-lo na Ferramenta Case BR Modelo, vamos colocar a mão na massa desenvolvendo a atividade a seguir:

Exercício 1

Faça o mapeamento do Modelo ER ilustrado a seguir, desenvolva-o na ferramenta Case BR Modelo, de

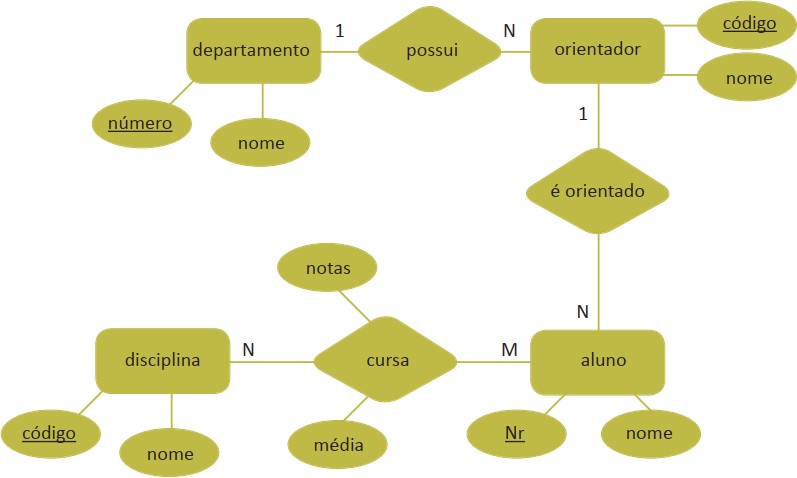


Imagem 32: A imagem demonstra um exemplo de DER para mapear os relacionamentos na ferramenta Br Modelo.

A entidade departamento tem com chave primária número e o atributo nome, que possui relação com a entidade orientador com chave primária código e atributo nome, com cardinalidade de 1 para N; a entidade orientador se relaciona com a entidade aluno com chave primária Nr e o atributo nome, com cardinalidade de 1 para N; a entidade aluno se relaciona com a entidade disciplina com chave primária código e o atributo nome, com cardinalidade de M para N. No relacionamento aluno com disciplina temos a relação cursa com os campos média e notas.

Exercício 2

A imobiliária João de Barro deseja manter controle dos condomínios pelos quais é responsável e nos passou as seguintes informações:

Um condomínio é formado por diversos apartamentos. Cada apartamento possui uma única garagem. Existe apenas um síndico responsável por cada condomínio. Deseja-se manter os dados dos proprietários de cada apartamento. Pode haver casos de um proprietário possuir mais de um apartamento. Com base nessas infor- mações, desenvolva o Modelo ER, definindo as entidades, a cardinalidade, as chaves e seus atributos, con- forme listados a seguir:

**Condomínio:** código, nome e endereço. **Apartamento:** número, tipo **Proprietário:** rg, nome, telefone, e-mail **Garagem:** número, tipo

**Síndico:** matrícula, nome, endereço, telefone

modelo,

construindo o Diagrama ER por meio de uma Ferramenta Case, como por exemplo o BR Modelo.

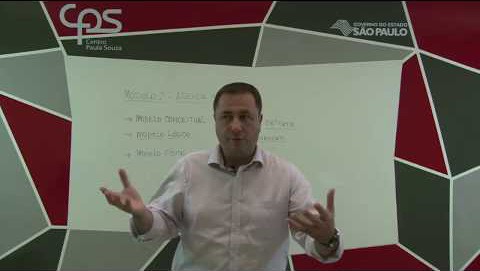
AMPLIANDO HORIZONTES



**Vídeo:**

Não deixe de assistir ao vídeo Modelo Entidade Relacionamento – Parte II.

Disponível em **https:**[**//www.youtube.com/watch?v=m4ALccjL**](http://www.youtube.com/watch?v=m4ALccjL) **A**. Acessado em 20/09/2018.



Para aprofundamento dos temas discutidos nesta aula, seguem abaixo algumas dicas de livros que se rela- cionam com o conteúdo estudado. Estas dicas são muito importantes para você!

**Livros:**

DATE, C.J. Introdução a Sistema de Banco de Dados. 8 ed. Campus, 2013. ELMASRI, R.; NAVATHE, S. B. Sistemas de bancos de dados. 6 ed. Pearson, 2012. TAKAHASHI, M.; AZUMA, S. Guia mangá de banco de dados. 2 ed. Novatec: 2011.



# AGENDA 11

**PROJETO LÓGICO DE BANCO DE DADOS - NORMALIZAÇÃO**

Duplicação de dados

###### Para acompanhar o cenário do atual mercado empre- sarial, é fundamental que as organizações tenham sem- pre informações disponíveis que auxiliem as operações,

***a*** ***gestão*** ***e*** ***a*** ***tomada*** ***de*** ***decisão,*** ***a*** ***fim*** ***de*** ***calcular*** ***ris****D****c****u****o****p****s****l****,****icação* *de* *dados*

###### otimizar processos e melhorar os resultados.

***Para tanto, é preciso garantir que os dados armazenados sejam íntegros, consistentes e confiáveis para que possam ser analisados e transformados em informações precisas.***

***Dados redundantes desperdiçam espaço em disco e criam problemas de manutenção. Imagine que em um mesmo banco de dados você tenha uma entidade Funcionário e uma entidade Pessoa e que ambas armazenam o atributo nome. Perceba que aí temos um exemplo de redundância porque se você tem um funcionário que é uma pessoa, estará armazenando duas vezes a mesma informação dentro do Bando de Dados, em locais diferentes, concorda?***



?

O mapeamento, estudado na agenda anterior, já auxilia na passagem do modelo conceitual para o modelo lógico relacional de dados, buscando minimizar a redundância de dados. No entanto, é impor- tante ainda aplicar as regras de Normalização para garantir a integridade, auxiliando na simplificação dos atributos de uma tabela. As regras de Normalização colaboram significati amente para a estabili- dade do modelo de dados e redu em consideravelmente as necessidades de manutenções.



A



A redundância de dados é a causa de vários problemas em esquemas de Banco de Dados Relacionais. Pode trazer transtornos com armazenamento, duplicidade de armaze- namentodedados,anomaliasdeinserção,deexclusãoede atualização.

z

v



Veja o que aconteceu com um esquema que Tânia proje- tou para a clínica médica da sua prima:

Ela criou uma entidade “Pacientes” com os seguintes atributos: Codigo, Nome, Endereco, Telefone, DataNasci- mento, SiglaConvenio, NomeConvenio, EnderecoConvenio, Telefone\_Convenio.

Você percebeu que Tânia projetou os dados de pacientes e os de convênios na mesma tabela? Isso que ela fez é um exemplo de um mau projeto!

Sabe por quê?

Porque os dados de um convênio (nome, endereço e telefone do convênio) são repetidos para cada paciente associado a esse convênio.

Veja, os dados de um determinado convênio, o “Boa Saúde”, por exemplo, serão repetidos para cada um de seus associados.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Código Nome** | | **Endereço** | **Telefone** **Data**  **de Nasc.** | **Sigla do Convê**  **nio** | **Nome do Convênio** | **Endereço Convênio** | **Telefone Convêni o** |
| 100 | Sonia deLucca | Av. José Brejá, 700 | 3333-8899 | BS | Boa Saúde | R.Arajá, 50 | 3322- |
|  |  |  | 10/08/1975 |  |  |  | 5588 |
| 200 | LaerteSouza | Av. Costa, 85 | 3333-5566 | BS | Boa Saúde | Rua Arajá, | 3322- |
|  |  |  | 20/07/1983 |  |  | 50 | 5588 |
| 300 Maria | | R.Castro Alves, 200 | 3232-9966 | BS | Boa Saúde | R. Arajá, 50 | 3322- |
| JoséPereira | |  | 30/04/1986 |  |  |  | 5588 |
| 400 | Dirce MariaSilva | R. Napoli, 400 | 3232-5580 | APS | Ass. Paulista de | R. Amapá, | 2745- |
|  |  |  | 17/06/1968 |  | Saúde | 340 | 3030 |
| 500 | JoãoMarques | R.15 de Novembro, | 3456-9090 | HSB | Hospital São ento | R. Trelo, 40 | 3535- |
|  |  | 80 | 20/08/2004 |  |  |  | 9060 |

Veja os **problemas que a redundância de dados** pode trazer:

B

*Anomalia deinserção:*

* Quandoseinserirumpacienteéprecisoinserirtambémosdadosdoconvênio,mesmoquejáestejamcadas- trados.
* Não é possível inserir um convênio sem inserir também umpaciente.

*Anomalia deexclusão:*

* Aoseexcluirumpaciente,seesteforoúnicoassociadodeumconvênio,então,osdadosdoconvênioserão perdidos.

*Anomalia demodificação:*

Quando for necessário modificar os dados de um convênio, é preciso atualizar os mesmos dados em todas os registros de pacientes que estejam associados àquele convênio.



Pense, que sufoco!

Para que você não corra o risco de cometer os mesmos erros da Tânia, é importante estudar as regras de normalização. Vamos lá?

Texto adaptado de **https:**[**//www.gsigma.ufsc.br/~popov/aulas/bd1/normalizacao\_old.html**.](http://www.gsigma.ufsc.br/~popov/aulas/bd1/normalizacao_old.html)

Acessado em 21/09/2018.



Normalização é um processo utilizado, após o mapeamento do modelo conceitual, para acertar possíveis problemas estruturais das entidades e relacionamentos com campos criados – também chamados de anomalias

– em um modelo de entidade erelacionamento. A Normalização consiste na análise dos atributos das entidades e relacionamentos com campos, sob o ponto de vista das regras chamadas formas normais, que descrevem, com base na teoria de conjuntos, na álgebra e no cálculo relacional, o que devemos ou não fazer nas estruturas das entidades e relacionamentos de nosso mod- elo, baseados em conceitos matemáticos.

Essa análise pode demonstrar a necessidade de alterarmos a estrutura de nossas entidades e relacionamentos com campos, dividindo ou agrupando seus atributos para aprimorar o processo de recuperação das informações (performance) e seu armazenamento, de modo a evitar perda, redundância e distorção da informação.

Vamos estudar as formas normais que nos auxiliarão nesse trabalho:

**1ª Forma Normal**

Remove grupos de repetição. É anormalização do registro, de forma que o relacionamento entre a sua chavee seus atributos seja unívoca, isto é, para cada chave há a ocorrência de um e somente um dado de cada atributo. Portanto, na 1ª Forma Normal, tratamos os atributos multivalorados.

*Passos da 1ª Forma Normal*

1. Identifique atributos que possuEm valores para uma ocorrência daentidade.
2. Remova os atributos com uma cópia da chaveprimária.

*Exemplo:*

Cliente (Número, Nome, {EnderecoEntrega}) => Não está na 1ª Forma Normal!

|  |  |  |
| --- | --- | --- |
| **Número** | **Nome** | **Endereço-Entrega** |
| 124 | João dos Santos | Rua 9 de Julho, 56 |
| 256 | José Ferreira | Av. 15 de Novembro, 1980  Av. Campos Sales, 250 |
| 311 | André Alves | Av. São Carlos, 95 Rua Jorge Assef, 900 Rua Rui  Barbosa,935 |

Conversão para 1FN

Cliente (Número, Nome, End\_Entrega)

|  |  |  |
| --- | --- | --- |
| **Número** **Nome** **Endereço\_Entrega** | | |
| 124 | João dos Santos | Rua 9 de Julho, 56 |
| 124 | João dos Santos | Av. 15 de Novembro, 1980 |
| 256 | José Ferreira | Av. Campos Sales, 250 |
| 311 | André Alves | Av. São Carlos, 95 |
| 311 | André Alves | Rua Jorge Assef, 900 |
| 311 | André Alves | Rua Rui Barbosa, 935 |

**2ª Forma Normal**

É a normalização do registro de forma que, já submetido à 1FN, apresente uma chave

Concatenada que se relacione de forma integral com todos os seus atributos – Dependência Funcional Total.

A relação está na 2FN se está na 1FN e se não existir atributo não chave que é dependente de só uma parte de qualquer chave candidata.

*Passos da 2ª Forma Normal*

1- Identifiqueatributosdependentessomentedepartedachaveprimária–DependênciaFuncionalParcial. 2- Remova os atributos encontrados com uma cópia de parte da chaveprimária.

*Exemplo:*

Pedido (Nr\_Ped, Data\_Pedido, Nr\_Peça, Descrição\_Peça, Quantidade\_Comprada, Preço\_Cotado) Não está na 2ª Forma Normal, pois:

A data do pedido depende apenas da chave do Número do Pedido A Descrição da peça depende apenas da chave Número da Peça

Portanto, esses atributos dependem de apenas uma parte da chave primária.

**Nr\_Ped => Data\_Pedido Nr\_Peça => Descrição\_Peça**

Só dependem de uma parte da chave primária.

**Anomalias de Atualização (Não está na 2ª Forma Normal)**

*Pedido*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nr\_Ped** | **Data** | **Nr\_Peça** | **Descrição** | **Qtd\_Comprad a** | **Preço\_Cotad o** |
| 1000 | 14/11/2018 | AX12 | Bicicleta | 3 | 100,00 |
| 1020 | 15/11/2018 | BT04 | TV | 10 | 400,00 |
| 1030 | 15/11/2018 | BZ66 | Bola | 300 | 10,00 |
| 1040 | 16/11/2018 | BT04 | TV | 4 | 390,00 |
| 1050 | 17/11/2018 | CB03 | Video-Game | 5 | 380,00 |
| 1070 | 20/11/2018 | BT04 | TV | 2 | 410,00 |

Veja os problemas que essa tabela traz:

*Gasto de Espaço de Armazenamento e anomalias*

**Atualização**

* Mudar descrição de BT04 implica em várias mudanças(tempo!)

*Dados Inconsistentes*

* A Peça BT04 pode ter descriçõesdiferentes.

*Adições*

- NãoserápossíveladicionarumanovapeçasemumpedidoparaelaporqueoNúmerodoPedido(Nr\_Ped)é chaveprimária!

*Eliminações*

* Se eliminar o Pedido 1000 perde-se a informação de que a peça AX12 ébicicleta.

*Para corrigir o Problema:*

* + Incluir nas relações os atributos correspondentes à Chave Primária apropriada (coleção mínima daqual dependem)
  + Atribuir um nome para cadarelação

*2ª Forma Normal*

*53*

**Pedido (Nr\_Ped, Data) Peça (Nr\_Peça, Descrição)**

**Linha\_Pedido (Nr\_Pedido, Nr\_Peça, Quantidade\_Comprada, Preço\_Cotado)**

PROJETO LÓGICO DE BANCO DE DADOS - PARTE 2

o

**3ª Forma Normal**

Uma relação está na 3FN se e somente se estiver na 2FN e todo atributo não chave não é transitivamente dependente de qualquer outro não chave: **Dependência Funcional Transitiva**.

*Passos para a 3ª Forma Normal:*

1. Identifique atributos dependentes de outros atributos nãochave.
2. Remova esses atributos com uma cópia do atributo do qualdepende.

*Exemplo:*

Cliente (Nr\_Cli, Nome, Endereço, Nr\_Vendedor, Nome\_Vendedor)

Não está na 3FN, pois “Nome do Vendedor” é transitivamente dependente de “Nr\_Vendedor” que por sua vez não é chave.

Para corrigir o Problema:

*3ª Forma Normal*

**o**

**Cliente (Nr\_Cliente, Nome, Enderec ) Vendedor (Nr\_Vendedor, Nome\_Vendedor)**

VOCÊ NO COMANDO



Quais os problemas em ter esse tipo de atributo?

matrícula

código

projeto

empregado

empregado

tipo

descrição

data

admissão

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ProjCodi go** | **ProjTipo** | **ProjDescrica o** |  | **Emprega do** | | |
|  | Novo Desenvolvime nto |  | **EmpMatrícul** | **EmpNom** | **EmpDataAdm** | **EmpDataTermPr** |
| LSC001 | Siste | **a** | **e** |  | **oj** |
| 1215 | Edmundo | 30/05/2010 | 24/09/2018 |
|  | ma |
| 3560 | Daniel | 18/09/2016 | 24/09/2018 |
|  | Contá |
| 5689 | Thiago | 02/09/2017 | 24/09/2018 |
|  | bil |
| 2156 | Karla | 20/08/2015 | 24/09/2018 |
| PAG02 | Manutenção | Contr le  de Estoque | 7856  4561 | Giovani  Marisa | 18/07/2016  08/03/2017 | 21/09/2018  21/09/2018 |

Observe atentamente a tabela com os dados. Vamos analisá-la para saber se você acertou! De acordo com o modelo acima:

Projeto (ProjCodigo, ProjTipo, ProjDescricao (EmpMatricula, EmpNome, EmpDataAdmissao))

*Verificando se o modelo está na 1ª Forma Normal:*

Para que a relação fique na 1ª Forma Normal, cada atributo composto ou multivalorado deve ser transformado em uma nova tabela e deve-se adicionar uma chave estrangeira nessa nova tabela.

Ficando dessa forma:

Projeto (ProjCodigo, ProjTipo, ProjDescricao)

Empregado (EmpMatricula, ProjCodigo, EmpNome, EmpDataAdmissao)

*Agora analise se essa tabela está na 2ª Forma Normal:*

Alocação

|  |  |  |  |
| --- | --- | --- | --- |
| EmpMatricul | ProjCodig | QtdeHorasTra | ProjNome |

Para saber se você acertou, acompanhe essaexplicação:

Paraavançarparaa2FNea3FNénecessárioconheceroconceitodedependênciafuncionalparaidentificar seoagrupamentodeatributosdeumatabelaéapropriado,evitandoredundânciadedados,inconsistênciase perda de dados em operações de remoções oualterações.

Existem 3 tipos de Dependência Funcional:

Dependência Funcional Total Dependência Funcional Parcial Dependência Funcional Transitiva

*Dependência Funcional Total:*

Osatributosnãochavedeumatabelatêmquedependerdachaveprimáriaesomentedela.Porexemplo:Uma determinada tabela possui sua chave primária composta pelos atributos A e B. Logo, C será dependente funcion- al total se e somente se C depender funcionalmente de A eB.

Levando para o nosso exemplo:

Alocação

Nesse caso, a quantidade de horas trabalhadas refere-se às horas trabalhadas por um empregado em um determinado projeto, portanto o atributo “QtdeHorasTrab” depende tanto da chave “EmpMatricula” quanto da chave “ProjCodigo”.

|  |  |  |  |
| --- | --- | --- | --- |
| EmpMatricul | ProjCodig | QtdeHorasTra | ProjNome |

*Dependência Funcional Parcial:*

Os atributos não chave de uma tabela dependem de parte da chave primária. Exemplo: Uma determinada tabela possui sua chave primária composta pelos atributos A e B. Logo, C será dependente funcional parcial se e somente se C depender funcionalmente de A ou B.

Levando essa definição para o nosso exemplo:

Alocação

Nesse caso, o nome do projeto está diretamente relacionado com o código do projeto e não tem nenhuma relação com o número da matrícula do funcionário, que é a outra chave da tabela. Portanto, sua dependência é parcial.

|  |  |  |  |
| --- | --- | --- | --- |
| EmpMatricul | ProjCodig | QtdeHorasTra | ProjNome |

A solução para a dependência funcional dessa tabela é separar em duas tabelas, conforme as dependências funcionais dos seus atributos:

Alocação Projeto

ProjNome

ProjCodigo

|  |  |  |
| --- | --- | --- |
| EmpMatricul | ProjCodig | QtdeHorasTra |

Agora, então, vamos analisar se a relação apresentada a seguir está na 3ª Forma Normal, verificando se há dependência funcional transitiva.

*Dependência Funcional Transitiva:*

O atributo C é dependente funcional transitivo de A se C é funcionalmente dependente de B e B funcional- mente dependente de A, na mesma tabela.



Empregado

Nesse caso, o atributo “DataTerminoProj” é dependente funcional do “CodProjetoEmp” que é dependente funcional do atributo “EmpMatricula”.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| EmpMatric | EmpNom | EmpDataAdmissa | CodProjetoEmp | DataTerminoPr |

A solução para a dependência funcional dessa tabela é separar em duas tabelas, conforme as dependências funcionais dos seus atributos:

Empregado

|  |  |  |  |
| --- | --- | --- | --- |
| EmpMatricu | EmpNom | EmpDataAdmissa | CodProjetoEmp |

Projeto

DataTerminoProj

CodProjetoEmp

Para os exercícios abaixo, realize o processo de normalização (1FN, 2FN e 3FN):

1. Levandoemconsideraçãoqueofuncionáriopodetervárioscargoseváriosdependentes,normalizeatabela a seguir para que sejam armazenados os dados cadastrais dos funcionários daempresa:

|  |  |  |
| --- | --- | --- |
| **Dados Cadastrais do funcionário** | | |
| Matrícula: | Nome: | |
| Data Nasc.: | Nacionalidade: | Sexo: |
| Est. Civil: | RG: | CPF: |
| Endereço: | Telefone: | Data de Adimissão: |
| **Cargos Ocupados** | | |
| Cargo: | Data Início: | Data Fim: |
| Cargo: | Data Início: | Data Fim: |
| **Dependentes** | | |
| Nome: | Data Nasc.: |  |
| Nome: | Data Nasc.: |  |

1. Normalizeatabelaaseguircomoobjetivodearmazenarosdadosdasfichasmédicasdetodosospacientes de uma clínica. Leve em consideração que o CRM do médico serve para identificá-lounicamente.

|  |  |  |
| --- | --- | --- |
| **Ficha Médica** | | |
| Número paciente: | Nome: | |
| Data Nasc.: | Sexo: | Convênio: |
| Est. Civil: | RG: | Telefone: |
| Endereço: | | |
| **Consult** **as** | | |
| Número consulta: | Data: Médico: Diagnóstico: | |
|  |  | |
|  |  | |
| **Exames** | | |
| Número consulta: | Exame: Data: | |
|  |  | |
|  |  | |

Adaptado de **https://docente.ifrn.edu.br/nickersonferreira/disciplinas/programacao-com-acesso-a-banco- dedados-3o-ano/lista-de-exercicios-01-normalizacao**. Acessado em 21/09/2018.

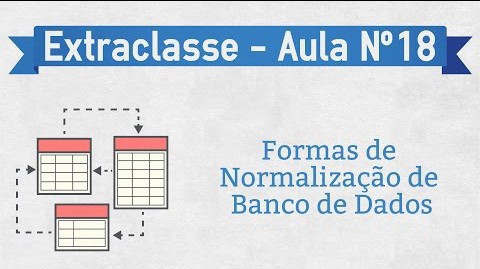
AMPLIANDO HORIZONTES



*Vídeos:*

“Formas de Normalização de Banco de Dados” - Aprenda a organizar suas tabelas de forma a garantir que seus dados sejam salvos de maneira correta sem perda de informação o redundância.

Disponível em **https:**[**//www.youtube.com/watch?v=XPj82D6G3lI**.](http://www.youtube.com/watch?v=XPj82D6G3lI) Acessado em 24/09/2018.



Esse vídeo traz uma revisão geral do estudo que fizemos nesse semestre sobre Banco de Dados I. Assista à vídeo aula da Prof. Caroline Milone do Centro Paula Souza.

Disponível em **https:**[**//www.youtube.com/watch?v=WF-BKoROE.**](http://www.youtube.com/watch?v=WF-BKoROE) Acessado em 24/09/2018.



Para aprofundamento dos temas discutidos nesta aula, seguem algumas dicas de livros que se relacionam com o conteúdo estudado. Estas dicas são muito importantes para você!

*Livros:*

ELMASRI, R.; NAVATHE, S. B. Sistemas de bancos de dados. 6 ed. Pearson, 2012. TAKAHASHI, M.; AZUMA, S. Guia mangá de banco de dados. 2 ed. Novatec: 2011. DATE, C.J. Introdução a Sistema de Banco de Dados. 8 ed. Campus, 2013

#### Referências Bibliográficas

Date, C.J. Introdução a Sistema de Banco de Dados. 8 ed. Campus, 2013.

DevMedia.ArtigoPadronizaçãodenomenclatura-RevistaSQLMagazine100.Disponívelemhttps://www.devme- dia.com.br/padronizacao-de-nomenclatura-revista-sql-magazine-100/24710. Acessado em13/09/2018.

Elmasri, R.; Navathe, S. B. Sistemas de bancos de dados. 6 ed. Pearson, 2012.

Ferreira, Nickerson. Programação com Acesso a Banco de Dados. Disponível em https://docente.ifrn.edu.br/nick- ersonferreira/disciplinas/programacao-com-acesso-a-banco-dedados-3o- ano/lista-de-exercicios-01-normalizacao. Acessado em 21/09/2018.

GSIGMA – Grupo de Sistemas Inteligentes de Manufatura da UFSC. Aula 4 Normalização do Esquema Relacional.

Disponível em

https:/[/www.gsigma.ufsc.br/~popov/aulas/bd1/normalizacao\_old.html.](http://www.gsigma.ufsc.br/~popov/aulas/bd1/normalizacao_old.html) Acessado em 20/09/2018.

Matos, Alessandro Nicoli. Informação é prata. Compreensão é ouro, 2010. Disponível em [http://plus.google.](http://plus.google/) com/118410405608511378703. Acessado em 12/09/2018.

Piva, Gustavo Dibbern. Informática, Análise e Gerenciamento de Dados. Manual de Informática Centro Paula Souza,

v. 3. São Paulo: Fundação Padre Anchieta, 2010.

Takahashi, M.; Azuma, S. Guia mangá de banco de dados. 2 ed. Novatec: 2011. Imagens: freepik, arquivos GEEaD