



BANCO DE DADOS

AULA 1

@explicadoresnet

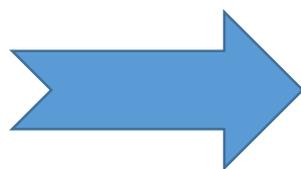


FICHA DE INSCRIÇÃO PROJETO SOCIAL

Informes gerais:

Inscrições abertas ao público: 04 de fevereiro a 08 de março
Período de aulas: 11 de março a 30 de novembro
Recesso em julho (período a definir)

Nome do Laboratório:			
Nome Completo:			
Telefone:		Email:	
Endereço:		Nº	Bairro:
CEP:	Cidade:	Estado:	
Data de Nascimento:	RG:	CPF:	
Naturalidade:			
Nome do Responsável:			Telefone:
Profissão:			
Escolaridade: Grau: 1ª () 2ª () 3ª () Completo () Incompleto ()			
É sócio do Clube de Espectadores? () sim () não Se não: deseja se inscrever? () sim () não			
Você participa ou já participou de atividades comunitárias e sociais? Como e quais:			
Qual seu meio de transporte prioritário para chegar aos laboratórios?			
carta de intenção			



Banco de Dados

“Um **dado** é um fato conhecido que pode ser registrado e possui significado implícito.”

“Um **banco de dados** é uma **coleção de dados** relacionados.”

“Um **banco de dados** é uma coleção logicamente coerente de dados com algum **significado inerente**.”



Banco de Dados

“Um **banco de dados** é algo mais complexo que um simples arquivo.”

O banco não possui apenas dados, mas também toda a estrutura que foi definida para o armazenamento das informações, o que chamamos de **catálogo**.

- ✓ Base de Dados
- ✓ Sistema Gerenciador de Banco de Dados
- ✓ Linguagem de Exploração
- ✓ Programas voltados para necessidades objetivas



Banco de Dados

Além dos quatro itens que compõem um Banco de Dados (software e dados), devemos sempre considerar:

“Um **hardware** capaz de abrigar todos esses dados.”

“O fator **pessoal** envolvido com o ambiente, isto é, seus **usuários**, que podem realizar eventos e solicitações a um banco de dados (**transação**).”



Banco de Dados

“Uma **base de dados** é uma coleção de arquivos que compõe as informações sobre alguma área e ficam armazenados em um meio físico.”

“Um **SGBD** vai ficar responsável por manipular os dados armazenados em uma base e evitar grandes listagens com informações não-requeridas.”

“As **Linguagens de Exploração** geralmente fazem parte do próprio SGBD, mas também podem ser externas. Juntamente com os **programas**, elas permitem atender solicitações de forma rápida e objetiva. ”



O Que É uma Base de Dados?

Uma base de dados é um local, ou espaço, onde informações estão armazenadas e de onde elas são recuperadas. Uma base de dados terá um nome, e este nome deverá representar o que aquela base armazena.

O Que É um Sistema de Banco de Dados?

Computacionalmente, um Sistema de Banco de Dados é uma ferramenta que será utilizada para armazenar informações. Essa ferramenta possui três principais características:

- Armazenar os dados.
- Relacionar os dados armazenados.
- Recuperar os dados rapidamente.



O Que É um Sistema Gerenciador de Banco de Dados (SGBD)?

Um SGBD é uma ferramenta muito mais completa que um Sistema de Banco de Dados. Um SGBD disponibiliza uma série de funcionalidades que permitem controlar e acompanhar melhor os dados armazenados.

As principais características de um SGBD são:

- Permitir o acesso concorrente às bases de dados;
- Realizar o gerenciamento de transações;
- Permitir criar e aplicar regras de segurança às bases de dados;
- Permitir criar regras que garantam a integridade da base de dados.



Acesso Concorrente

O acesso concorrente à base de dados significa que o SGBD permite que duas ou mais pessoas acessem uma mesma base de dados ao mesmo tempo e o sistema controla para que um acesso não interfira no outro.

Gerenciamento de Transações

Uma transação em banco de dados consiste em um conjunto de operações que é tratado como uma unidade lógica indivisível. Isso significa que quando começa a execução de uma transação, esta deve ter executadas todas as operações dentro dela. Se acontecer qualquer falha durante a execução da transação (por exemplo: falta de energia, alguém desligar o servidor, cancelamento da transação pelo usuário, etc.) as operações pendentes devem ser canceladas, e aquelas que foram executadas deverão ser desfeitas. Isso acontece para garantir a integridade dos dados dentro da base.



Regras de Segurança

Outro aspecto importante com o qual o SGBD se preocupa é com a segurança das informações armazenadas. Sendo assim, um SGBD tem mecanismos para criação de regras de segurança.

As regras de segurança vão desde a definição de login e senha para os usuários, até a permissão de acesso ao SGBD e acesso aos dados armazenados.

Regras de Integridade

Outra característica importante de um SGBD a se destacar é a possibilidade de criação de regras que garantam a integridade da base de dados.

As regras de integridade são interessantes porque ficam definidas para uma base de dados, e todas as aplicações que acessaram aquela base poderão utilizar a regra de integridade.

Além das quatro características citadas e explicadas acima, os SGBD possuem várias outras funcionalidades como, por exemplo, algoritmos para extração de conhecimento em base de dados (data mining), técnicas OLAP e de definição de data warehouse, etc.

Atualmente, os SGBD funcionam em uma arquitetura cliente-servidor. Isso significa que você pode instalar o SGBD em um servidor e instalar o cliente em várias máquinas para que vários usuários tenham acesso simultâneo ao SGBD. É importante ressaltar que nessa arquitetura todas as bases de dados são criadas, alteradas e excluídas no servidor e não na máquina do cliente.



Fases no Desenvolvimento de um Projeto de Banco de Dados

Um projeto de banco de dados compreende as seguintes fases:

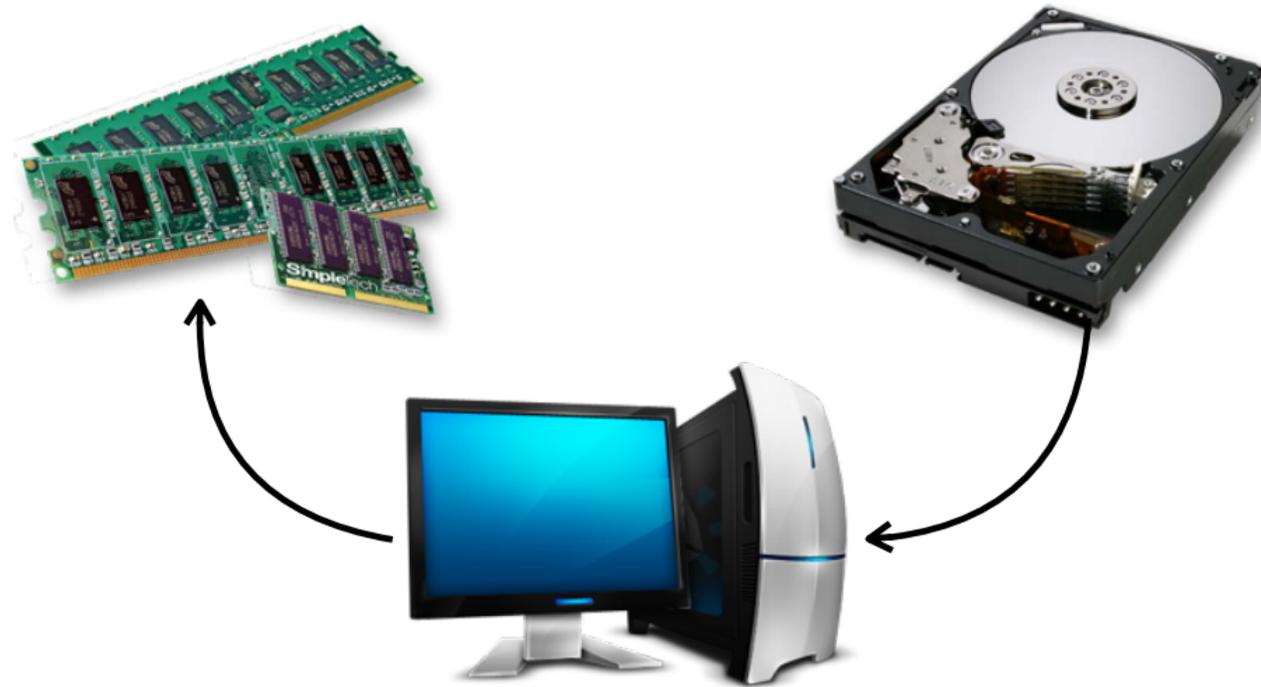
- **Modelagem Conceitual:** A modelagem conceitual refere-se ao desenvolvimento de um modelo inicial da base de dados que reflita as necessidades do usuário. Essa modelagem preocupa-se em descrever quais dados serão armazenados na base de dados e quais dados se relacionam.
- **Modelagem Lógica:** A modelagem lógica compreende o processo de descrever como os dados serão armazenados no sistema e como irão se relacionar. Isso significa transformar o modelo conceitual obtido na primeira fase num modelo mais próximo da implementação.

Implementação do Modelo Lógico: Uma vez que toda a etapa de modelagem esteja concluída, será necessário implementar ou criar a base de dados no SGBD escolhido. Essa fase requer que o desenvolvedor conheça a Linguagem SQL e conheça o SGBD selecionado.

A fase de modelagem é a principal etapa no desenvolvimento de uma base dados. Por isso é muito importante que se dedique tempo e esforço no desenvolvimento de uma boa modelagem da base de dados.



Memória vs. Veículos



Dados e Arquivos

“**Dados** são armazenados em **arquivos**.”

“**Arquivo** é um conjunto de informações referentes aos elementos de um **conjunto** (programas ou dados).”

“**Arquivos** são áreas reservadas em qualquer **dispositivo de memória** para a inclusão de informações a qualquer momento.”



Arquivos de Entrada e Saída

“**Arquivos de Entrada** são aqueles armazenados em qualquer veículo de **leitura** (CD, DVD, etc)”

“**Arquivos de saída** podem ser impressos como **relatórios escritos** ou residir em outros meios magnéticos/óticos.”



Registro Lógico e Físico

“**Registro** ou **registro lógico** é cada elemento bem definido do **arquivo**.”



“O **Registro físico** é a quantidade de informação transmitida à memória ou retirada dela em consequência de **uma única instrução**.”



Registros e Blocos

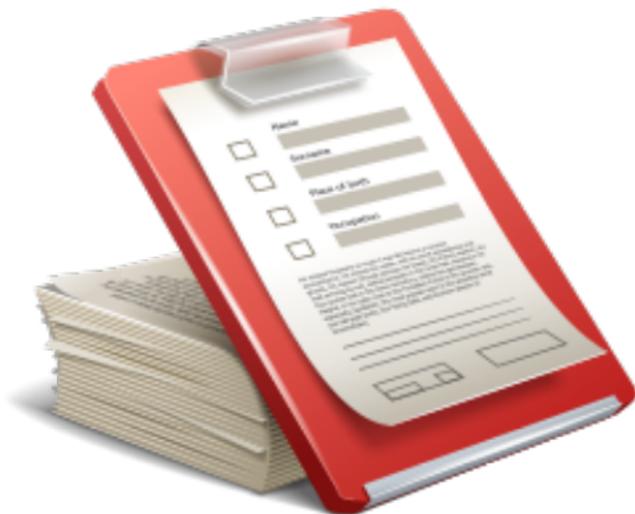
“Um **registro lógico** pode ser maior ou menor que um **registro físico**.”

“O mais comum é ter um **registro físico** que contém dois ou mais **registros lógicos**.”

“A relação entre o numero de **registros lógicos** e o numero de **registros físicos** se chama **fator de bloco** ou **blocagem**.”



Campos



“Os **campos** são espaços reservados aos diferentes dados que compõem um **registro**.”

“**Campos** correspondem a uma característica ou propriedade do objeto representado.”

“Todo **campo** tem um **tipo** (numérico, alfabético, alfanumérico) e um **tamanho**.”



Chave Primária

“Cada **registro lógico** normalmente tem um **campo chave primária** que o **individualiza.**”



“Uma **chave primária** pode ser formada por **vários campos.**”

“A **chave primária** também pode se chamar **chave de ordenação** ou **chave de classificação.**”

“Podemos ter **outras chaves** que não são primárias, mas podem ser usadas como **chave de ordenação.**”

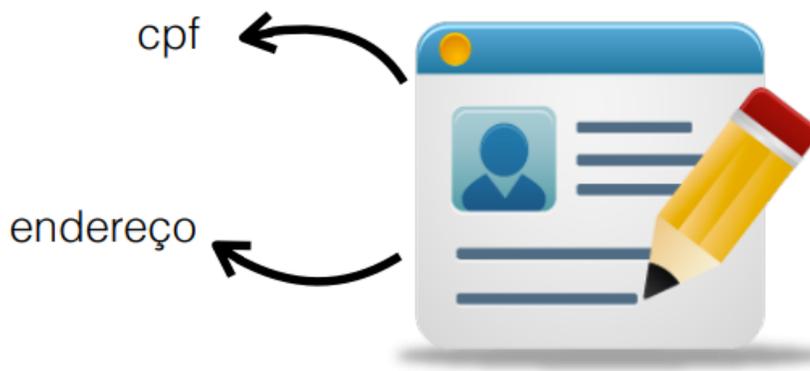


Chave Secundária



“Uma **chave primária** permite identificar **apenas um registro** dentro do **arquivo**.”

“Uma **chave secundária** permite identificar **vários registros** dentro do **arquivo**.”



Chave de Acesso

“Uma **chave de acesso** é usada para identificar os **registros** desejados em uma operação de **acesso**.”

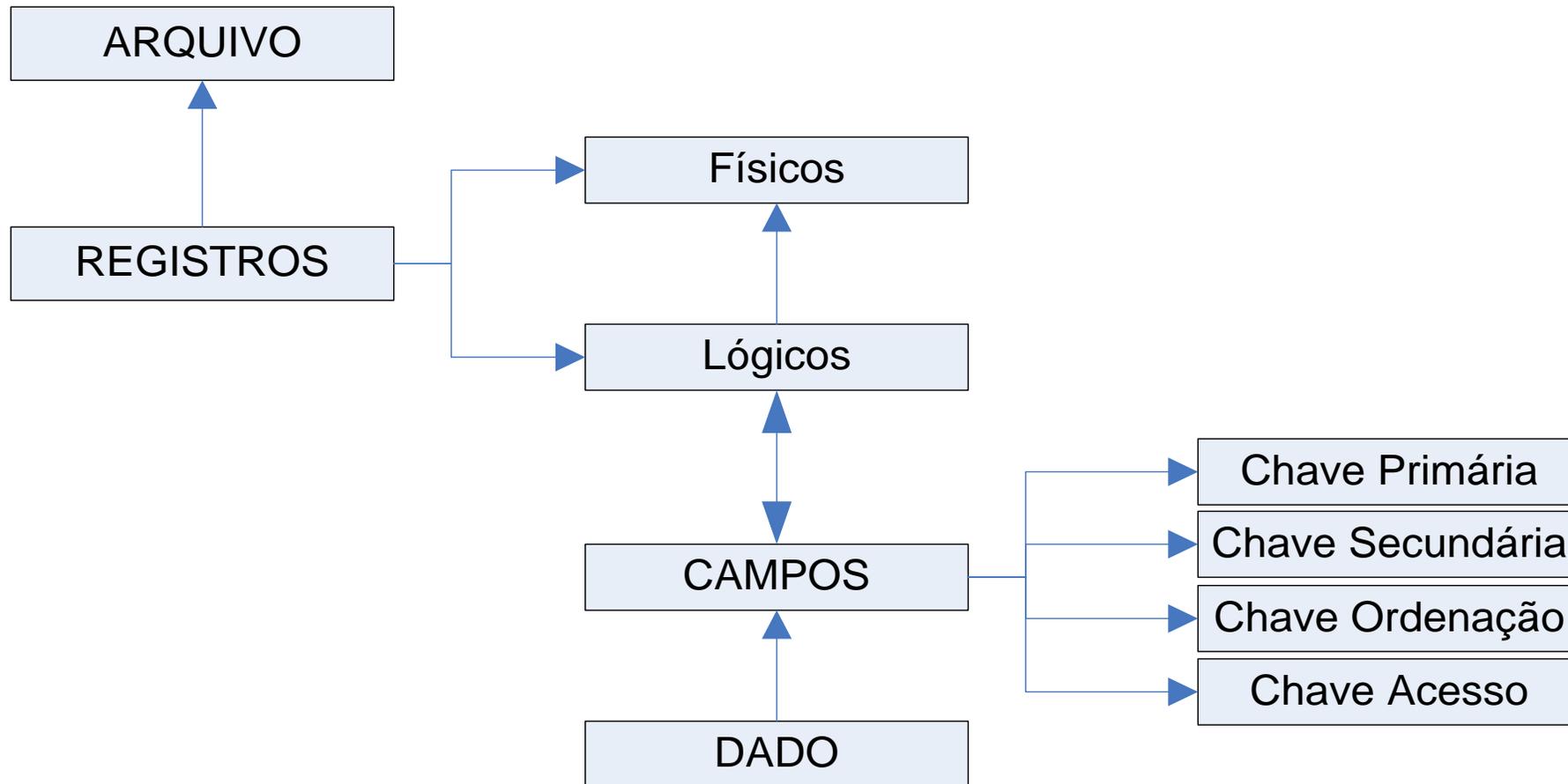


“O valor de uma **chave de acesso** em uma operação é chamado de **argumento de pesquisa**.”



Selecionar todos os registros dos cadastrados que moram em bairro = “Bangu”





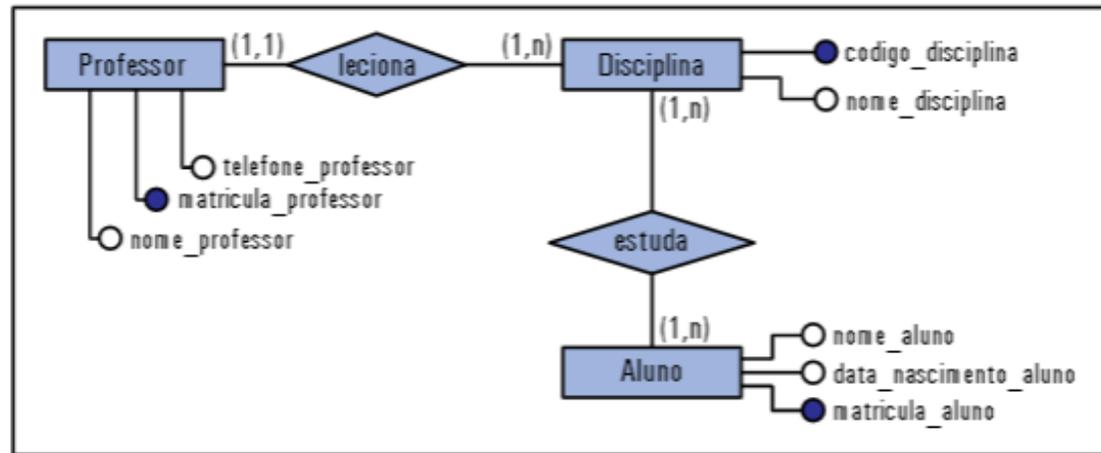
Modelo de Dados

O modelo da base de dados é também conhecido como Esquema da base de dados. Um esquema não deve mudar com frequência, porque uma vez alterado, tudo o que estiver envolvido com esse esquema (aplicações de usuários que acessem a base de dados, diagramas, etc.) terá de ser revisado e, muitas vezes, alterado.

O ideal é que haja independência dos dados em relação às aplicações que acessam aqueles dados. No entanto, essa independência é bastante difícil de alcançar, mas uma vez que alterada a estrutura (ou esquema) de uma base de dados, há a necessidade de se alterar também as aplicações que acessam aquela base.



Modelo de Entidade e Relacionamento



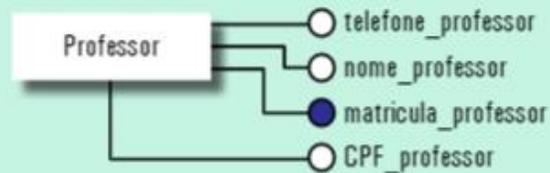
Uma entidade representa um conjunto de objetos do mesmo tipo do mundo real e sobre os quais se pretende armazenar dados.



Um atributo pode ser representado graficamente por uma elipse com o nome do atributo dentro da elipse. A elipse é ligada à entidade por uma linha, conforme exemplo:



Outra forma de representação utilizada por algumas ferramentas é representar o atributo como uma bolinha ligada à entidade e com o nome do atributo ao lado, conforme exemplo:



Uma entidade deve ter ao menos dois atributos. Uma entidade que possui apenas um atributo não é entidade e esse único atributo deveria estar em alguma outra entidade do modelo.

Todo atributo possui um tipo de dado que representa os valores permitidos para aquele atributo. A esse tipo de dados dá-se o nome de domínio do atributo



Tipos de atributos:

Atributo simples: é o atributo indivisível, que não pode ou não deve ser decomposto. Por exemplo: “CPF”, “numero da matrícula”, “RG”, “preço do produto”, etc.

Atributo composto: é o atributo que pode ser decomposto em outros atributos simples. Por exemplo, o atributo “endereço” pode ser decomposto em “nome da rua”, “número” e “complemento”.

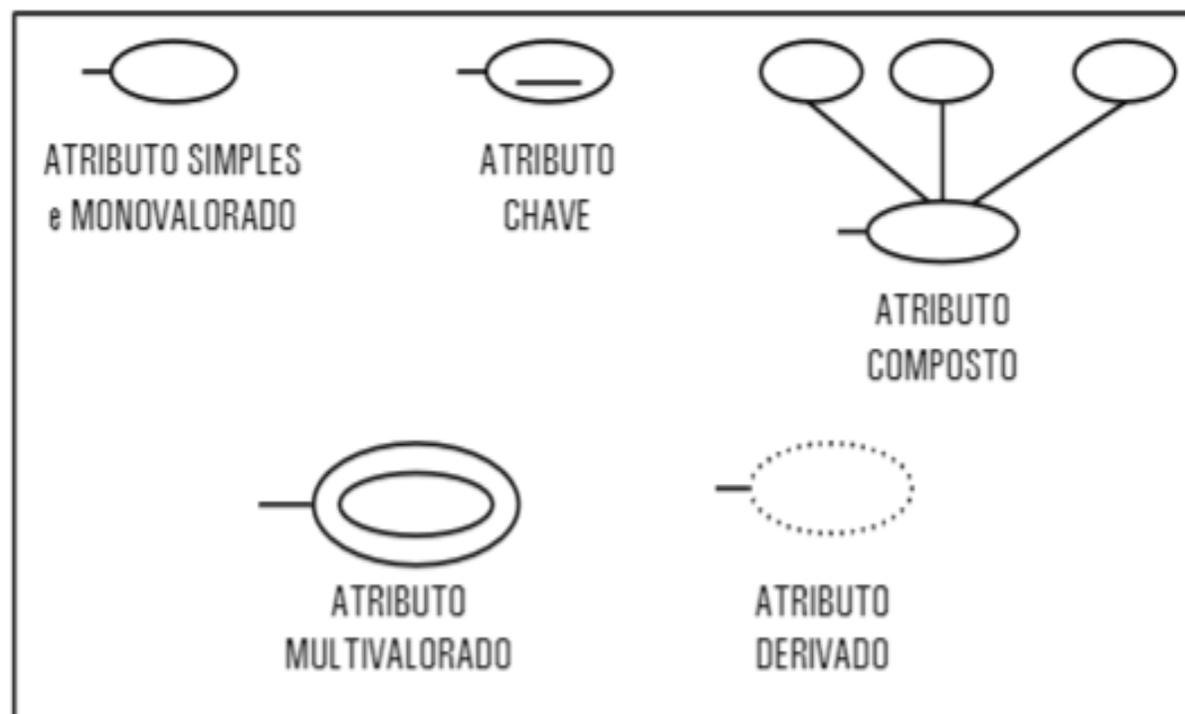
Atributo monovalorado: é o atributo que permite apenas o armazenamento de um valor por vez. Por exemplo, o atributo “CPF” é monovalorado porque uma pessoa possui apenas um número de CPF.

Atributo multivalorado: é o atributo que permite armazenar mais de um valor ao mesmo tempo no mesmo campo. Por exemplo, o atributo e-mail pode ser multivalorado uma vez que uma pessoa possui, normalmente, mais de um endereço de e-mail.

Atributo nulo: é o atributo que permite que seja inserido um valor nulo para ele.

Atributo derivado: é o atributo cujo valor para ele deriva de outro(s) atributo(s). Por exemplo, suponha que a sua entidade se chame compra e que ela tenha os seguintes atributos: “número da compra”, “data da compra”, “valor da compra”, “percentual de desconto” e “valor da compra com o desconto”.





OBS: É importante destacar que não existe mais de uma chave primária por entidade. Essa chave primária poderá ser simples ou composta.

Mesmo composta, é uma única chave primária composta de mais de um atributo.

Entidade Fraca

Entidade fraca é um tipo de entidade que não possui atributo chave primária por si só. Isso significa que não é possível definir uma chave primária, nem simples e nem composta, para a entidade.

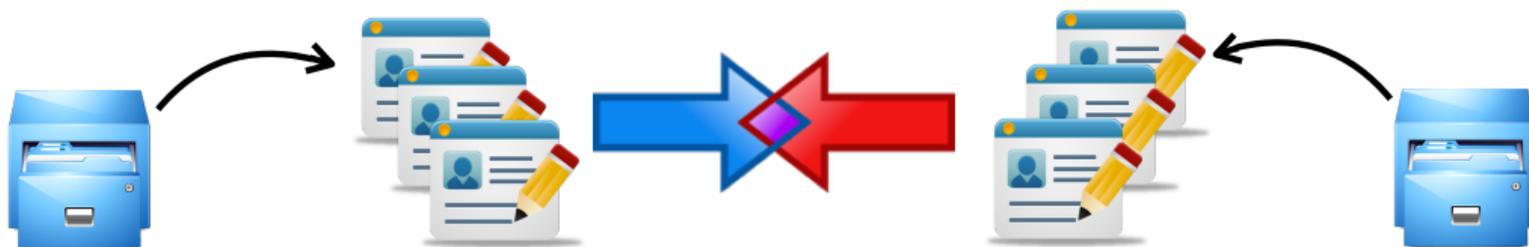


Ordenação e Intercalação



“A técnica de **sort** (ordenação) é usada para listar um cadastro segundo determinada **ordem** ou **classificação**.”

“A técnica de **merge** (intercalação) é usada para gerar um novo **arquivo único** a partir de dois ou mais **arquivos** que estão **classificados** segundo um **mesmo critério** pela **mesma chave**.”



Operações com Registros

Principais:

- ✓ **Inclusão**
- ✓ **Exclusão**
- ✓ **Pesquisa** (alterar/consultar)

Outras:

- ✓ **Atualização**
- ✓ **Reorganização**
- ✓ **Listagem total**
- ✓ **Sort**
- ✓ **Merge**
- ✓ **Append** (junção)



Normalização

O processo de normalização geralmente é aplicado quando temos uma base de dados que foi criada antes da existência de um banco de dados relacional ou foi desenvolvida sem considerar a existência de um banco de dados relacional.

Na maioria dos casos, esses sistemas são antigos e seus arquivos de dados possuem muitas informações redundantes e inconsistentes. Além disso, normalmente não existe nenhum documento que especifique o modelo de dados, o que torna muito difícil a manutenção desses sistemas ou as migrações para um banco de dados relacional.

Vale ressaltar que o modelo conceitual da base de dados agiliza bastante o processo de manutenção, porque permite que pessoas que não participaram do desenvolvimento do projeto possam entendê-lo mais rapidamente

Também podemos utilizar o processo de normalização para conferir se o nosso modelo de dados está normalizado e, caso não esteja, pode-se normalizá-lo antes da implementação da base de dados no SGBD. O processo de normalização consiste em um conjunto de regras, denominadas formas normais. A literatura apresenta 6 formas normais: 1FN, 2FN, 3FN, 4FN, 5FN e a de Boyce/Codd. Alguns autores tratam a forma normal de Boyce/Codd como sendo um caso específico da 3FN.



Código pedido	Valor Pedido	Data pedido	Telefone de contato	Produto				
				Código produto	Nome produto	Valor unitário do produto	Quantidade	Valor pago por produto
100	3300,00	30/11/2009	2222-2222 9999-9999	1	Computador	1500,00	1	1500,00
				5	Impressora	600,00	2	1200,00
				6	Papel A4	12,00	50	600,00
101	3800,00	15/12/2009	2121-2121 9191-9191	2	Mouse	30,00	10	300,00
				5	Impressora	600,00	5	3000,00
				7	Teclado	50,00	10	500,00

Primeira Forma Normal (1FN)

Toda tabela está na 1FN se os seus atributos forem atômicos. Isso significa que não serão permitidos atributos compostos, multivalorados ou grupos repetidos de dados (também conhecidos como tabelas aninhadas).



Segunda Forma Normal (2FN)

A 2FN só é aplicável para tabelas que possuem uma chave primária composta e que, além disso, tenham outros atributos que não façam parte da chave primária.

Uma tabela está na 2FN se estiver na 1FN e todo atributo que não compõe a chave primária deve ter dependência funcional total em relação à chave primária.

Terceira Forma Normal (3FN)

Uma tabela está na 3FN se estiver na 2FN e ela não possuir dependências transitivas

Dependência Transitiva ocorre quando existe um atributo que não é chave e nem faz parte da chave, mas que identifica outros atributos. Ou seja, existe um atributo não chave que depende de outro atributo não chave.

