

## Aula 14

### Processo de Normalização

#### Meta

Fornecer um apoio para o projeto de banco de dados relacional para a tarefa de normalização de um modelo de banco de dados.

#### Objetivos

Ao final desta aula, esperamos que você seja capaz de:

1. Compreender o conceito e importância sobre a Normalização.
2. Aplicar a Primeira Forma Normal (1FN).
3. Aplicar a Segunda Forma Normal (2FN).
4. Aplicar a Terceira Forma Normal (3FN).

#### Introdução

##### Porque devemos normalizar?

É necessário normalizar dados para gerenciar apropriadamente um banco de dados relacional. Consequentemente, é preciso dividir a tabela.

O processo de normalização consiste em regras de ouro em cada uma das etapas; o conceito principal é de "não misturar assuntos em uma mesma Tabela". Por exemplo: na Tabela Clientes devemos colocar somente campos relacionados com o assunto Clientes. Não devemos misturar campos relacionados com outros assuntos, tais como Pedidos, Produtos, etc. Essa "Mistura de Assuntos" em uma mesma tabela acaba por gerar repetição desnecessária dos dados bem como inconsistência dos dados. Normalmente após a aplicação das regras de **normalização de dados**, algumas tabelas acabam sendo divididas em duas ou mais tabelas, o que no final gera um número maior de tabelas do que o originalmente existente. Este processo causa a simplificação dos atributos de uma tabela, colaborando significativamente para a estabilidade do modelo de dados, reduzindo-se consideravelmente as necessidades de manutenção. Nesta aula, você irá estudar sobre esses conceitos importantes.

#### Fim da Introdução

## **O que será normalização?**

Normalização é o processo formal passo a passo que examina os atributos de uma entidade, com o objetivo de evitar anomalias observadas na inclusão, exclusão e alteração de registros.

Normalização de Dados consiste em definir o formato lógico adequado para as estruturas de dados identificados no projeto lógico do sistema, com o objetivo de minimizar o espaço utilizado pelos dados e garantir a integridade e confiabilidade das informações.

## **Normalização - Benefícios**

Dentre vários pontos como benefícios do processo de normalização vamos descrever alguns.

### **1. Estabilidade do Modelo Lógico**

- Entende-se por estabilidade a capacidade de um modelo manter-se inalterado diante a mudanças que venham a ser percebidas ou introduzidas no mini-mundo modelado.
- A implementação de tabelas não normalizadas compromete o próprio modelo lógico, ou seja, problemas nas estruturas de armazenamentos, alocação de espaços, etc. Afetando em menor ou maior grau o desempenho e a produtividade do desenvolvimento de aplicações.

### **2. Flexibilidade**

- Por flexibilidade entendemos a capacidade de adaptação a demandas diferenciadas, a expansão e redução ou omissão ou presença. Se numa estrutura de dados implementada, não tivermos a devida flexibilidade, estaremos impondo restrições de uso e tornando as estruturas de dados e de processos dependentes de limites rígidos.
- Através da flexibilidade => Aumenta o grau de adaptabilidade do banco de dados em função de mudanças no Mundo Real (Regras de Negócio e Requisitos de Informação)

### **3. Integridade**

- As estruturas de dados obtidas pelo processo de modelagem necessitam de recursos para que os dados a serem armazenados possam ter qualidade. A qualidade de um dado está vinculada: atualidade, veracidade, fidelidade, integridade, etc. A integridade diz respeito à qualidade do dado.

- Se um dado aparece mapeado em mais de um local de modo diferente, poderemos ter indícios de que não há integridade entre eles.
- Através do processo de normalização a não integridade (anomalia) é eliminada no modelo.

#### 4. Economia

- Os custos de armazenamento, gerados pela redundância, pela desnormalização e por outras anomalias presentes na construção das tabelas são os maiores custos existentes.
- Esse custo representa todo e qualquer esforço, tempo, ou valor agregado ao fato de manipulação volumes de dados maiores do que os efetivamente necessários.
- Tempos envolvidos em processamento, discos, mídias para backup, local para armazenamento e outros podem ter seus custos aumentados.
- Manter processos redundantes teremos custos adicionais como o caso de atualização dos dados.
- Através do processo de normalização, minimiza os custos de qualquer esforço, tempo e armazenamento.

#### 5. Expressividade

- Neste caso trata-se de um aspecto subjetivo, pois procura-se obter a máxima representatividade entre o ambiente observado e o modelo de banco de dados.
- Através do processo de normalização, aumenta o grau de representatividade do banco de dados tornando os "objetos do banco de dados" mais próximos dos objetos, respectivos, do Mundo Real.
- E também aumenta o grau de consistência das informações armazenadas no banco de dados.

## Formas Normais

A normalização é feita através da análise dos dados que compõem as estruturas utilizando o conceito chamado “Formas Normais (FN)”. As FN são conjuntos de restrições nas quais os dados devem satisfazê-las.

Exemplo: pode-se dizer que a estrutura está na primeira forma normal (1FN), se os dados que a compõem satisfazem as restrições definidas para esta etapa.

A normalização completa dos dados é feita seguindo as restrições das três formas normais existentes, sendo que a passagem de uma FN para outra é feita tendo como base o resultado obtido na etapa anterior, ou seja, na FN anterior.

Para realizar a normalização dos dados, é primordial que seja definido um campo chave para a estrutura, campo este que permite identificar os demais campos da estrutura. Toda entidade do banco de dados possui um atributo ou um conjunto de atributos do tipo chave primária já identificados durante a etapa de modelagem das entidades e relacionamentos, nesta forma veremos as diferentes formas de normalização.

### Primeira Forma Normal (1FN)

*Regra 1FN: Cada atributo de uma tabela deve ter apenas um único valor.*

Devemos,

1. transformar os atributos em atributos atômicos;
2. extrair com os grupos de dados repetidos.

Exemplo:

Como estudo, iremos considerar a tabela de Acessórios vendidos:

Código	Tipo	Forma	Marca	Material
001	brinco	losango	M&M	prata, ouro 18K
003	anel	redonda	Stylus1	ouro 18k, prata, diamante
005	efígie	retangular	R12Z	Ouro18k, rubi, perola
008	relógio	quadrado	Cincetti	prata, couro

Figura 12,1 Tabela Acessórios Vendidos

Para ser atômico, o atributo Material deveria conter apenas um dos materiais e não dois ou três materiais na mesma coluna. Desdobrando o atributo Material:

Código	Tipo	Forma	Marca	Material 1	Material 2	Material 3
001	brinco	losango	M&M	prata	ouro 18K	
003	anel	redonda	Stylus1	ouro18k	prata	diamante
005	efígie	retangular	R12Z	perola	ouro18k	
008	relógio	quadrado	Cincetti	prata	couro	

Figura 12.2 Tabela Acessórios com dados não atômicos

OK, resolvemos o problema de atributo atômico, mas note na tabela acima ainda não está na 1NF, pois veja, existem repetição de materiais nas próprias colunas de Materiais na tabela de acessórios e isso acontece porque cada tipo de acessório pode estar formado por mais de um material.

Aplicando a regra na 1FN, devemos retirar da tabela todos os grupos ou itens repetitivos e formar uma nova entidade ou tabela. Também devemos transportar a chave primária da entidade original para a nova entidade gerada.

Como resultado desta etapa, ocorre um desdobramento dos dados em duas tabelas, a saber:

Código	Tipo	Forma	Marca
001	brinco	losango	M&M
003	anel	redonda	Stylus1
005	efígie	retangular	R12Z
008	relógio	quadrado	Cincetti

Figura 12.3 Tabela Acessórios sem material

Código	Material
001	prata
001	ouro 18K
003	ouro18k
003	prata
003	diamante
005	perola
005	ouro 18k
008	prata
008	couro

Figura 12.4 Tabela Materiais

Observamos as tabelas criadas acima:

- Na figura 12,3 ficamos com os dados código, tipo, forma e marca.
- Na figura 12.4, ficamos com os dados código do acessório e material.
- O código do acessório deve existir em ambas as tabelas, para poder identificar se existem associações entre as duas tabelas.

==> Agora, nenhuma dessas tabelas possui grupos repetidos e cada atributo em ambas as entidades contém um valor simples ou atômico. Uma tabela que resulta de uma divisão como essa está na 1FN.

## Início da atividade

### Atividade 2 - Atende ao objetivo 2

A tabela a seguir mostra os diversos produtos de informática para cada pedido. Normalize essa tabela na 1FN:

Número Pedido	Código Fornecedor	Nome Fornecedor	Identidade	Endereço	Data Emissão	Produtos			
						Código	Nome	Qtde	Preço Unit.
02	012342	Casa SS	0231441-2	E 11 RJ	05/04/11	02221	DVD	100	0,99
						01341	CD-R	200	0,40
						04122	CD-RW	500	0,60
03	041211	Computer7	1411981-0	Bahia 82 SP	21/04/11	02711	Pendrive	500	12,99
						07107	Bateria	127	14,99
04	032329	RioPlus	3210062-1	Leste 231 SC	17/05/11	05123	Modem	300	15,99
						06410	Cargador	15	5,99

### Resposta Comentada

Note que cada pedido está representado por uma só linha, e as colunas de Produtos são um item de repetição onde aparecem vários produtos de um mesmo pedido. No caso do número de pedido 02 existem três linhas de produtos em uma única linha de dados. Isso porque em algumas vezes é preciso processar dois ou mais produtos num pedido. Claramente, observamos que esta tabela ainda não está na 1FN. Devemos:

1. Transformar os atributos em atributos atômicos: Analisando os atributos existentes podemos identificar alguns que podem merecer cuidados adicionais, Vejamos:

- O atributo Identidade será tratado de forma atômica, pois não será separado do dígito controlador.
- O atributo Endereço será desmembrado em Rua, Número e Estado, por se tratar de dados com naturezas completamente distintas.
- O atributo Data Emissão será tratado de forma atômica através do domínio DATE. Este é um tipo de dado especial em que toda a data é considerada como um atributo atômico.

Obtendo assim a primeira tabela:

Número Pedido	Código Fornecedor	Nome Fornecedor	Identidade	Rua	Número	Estado	Data Emissão	Produtos			
								Código	Nome	Qtde	Preço Unit.
02	012342	Casa SS	0231441-2	E	11	RJ	05/04/11	02221	DVD	100	0,99
								01341	CD-R	200	0,40
								04122	CD-RW	500	0,60
03	041211	Computer7	1411981-0	Bahia	82	SP	21/04/11	02711	Pendrive	500	12,99
								07107	Bateria	127	14,99
04	032329	RioPlus	3210062-1	Leste	231	SC	17/05/11	05123	Modem	300	15,99
								06410	Cargador	15	5,99

2. Extrair com os grupos de dados repetidos: Seguido pela explicação dada acima, devemos retirar da tabela todos os grupos ou itens repetitivos e com eles dar origem a novas linhas da tabela onde o conteúdo das demais colunas será o mesmo da linha original. Também devemos transportar a chave primária da entidade original para a nova entidade gerada. Fazemos isso e obteremos as seguintes tabelas:

Número Pedido	Código Fornecedor	Nome Fornecedor	Identidade	Rua	Número	Estado	Data Emissão
02	012342	Casa SS	0231441-2	E	11	RJ	05/04/11
03	041211	Computer7	1411981-0	Bahia	82	SP	21/04/11
04	032329	RioPlus	3210062-1	Leste	231	SC	17/05/11

Número Pedido	Código Produto	Nome Produto	Qtde	Preço Unit.
02	02221	DVD	100	0,99
02	01341	CD-R	200	0,40
02	04122	CD-RW	500	0,60
03	02711	Pendrive	500	12,99
03	07107	Bateria	127	14,99
04	05123	Modem	300	15,99
04	06410	Cargador	15	5,99

Observamos as tabelas criadas acima:

- Utiliza-se o campo Número de Pedido como atributo chave junto com o código de produto, pois não poderá haver o mesmo código de produto e número de pedido, portanto, trata-se de uma chave composta.

==> Nenhuma dessas tabelas possui grupos repetidos, cada atributo em ambas as entidades contém um valor simples ou atômico, então as tabelas acima estão na 1FN.

**Fim da atividade**

O intuito da 2FN é avançar, ainda mais, na direção de obter tabela(s) que não possua anomalias.

## Segunda Forma Normal (2FN)

*Regra 2FN: Além de atingir a 1NF, todos os atributos que não são chaves não dependem parcialmente dessa chave.*

Vejamos:

- Para que se aplique a regra da 2FN sobre uma tabela deve-se garantir, primeiramente, que ela esteja na 1FN. Caso contrário, não haverá sentido em se aplicar a 2FN.
- “...todos os atributos que não são chaves”, este texto deixa claro que devemos “excluir” de nossa análise as colunas formadoras da chave primária dessa entidade.
- “...não dependem parcialmente dessa chave”, deve-se perguntar a cada atributo, que não seja a chave, se ele depende apenas da chave da entidade. Caso contrário, devemos separar os atributos independentes e criar dentre os atributos separados uma nova chave para esta nova entidade. Essa chave deve ser mantida na entidade original como atributo de relacionamento entre ambas as entidades. Desta forma, não se perde qualquer informação no modelo.

Exemplo:

Imagine uma tabela chamada Prova\_Vestibular:

Código Curso	Código Aluno	Data de inscrição	Número da sala	Nome do Aluno
0123	01211131	03/05/2011	101	Lucas Santos
0222	03121457	02/05/2011	103	Maria Silva
0311	03210001	25/04/2011	101	Jonas Pereira
0311	15663011	30/04/2011	105	Sônia Castro
0425	01211131	04/03/2011	110	Lucas Santos

Figura 12.5 Tabela Prova-vestibular

Considere-se o código do curso e o código do aluno como chave primária, pois em alguns casos vários alunos realizaram a prova de vestibular para o mesmo curso e em outros casos um aluno realiza a inscrição para diferentes cursos; é claro que isto é possível se as provas de vestibular forem realizadas em datas diferentes.

Observamos que:

- A tabela já está na 1FN, mas ainda não, na segunda forma normal.
- Existe atributos que possuem dependência somente parte da chave. Ex:



→ Conhecendo o código do aluno podemos determinar seu nome mesmo que não soubermos o código do curso. O mesmo é válido para o número de sala onde será aplicado a prova.

→ Conhecendo o código do curso podemos também determinar o número da sala ainda que não saibamos o código do aluno.

*Então tanto o número da sala e o nome do aluno mostram a dependência parcial da chave primária. Elas não participam da chave primária da entidade Prova\_Vestibular. Devemos criar novas entidades e excluí-las da entidade original.*

- Existe atributos que possuem dependência total da chave. Ex:
  - Para determinar quando foi realizada a data de inscrição da prova devemos conhecer tanto o código do aluno e o código do curso, pois se tivermos o código do aluno podemos obter diversas datas de inscrição.
  - De igual forma, se conhecermos o código do curso podemos obter diferentes datas de inscrição, uma para cada curso onde o aluno se inscreveu.

Aplicando a regra 2FN visto as dependências parciais obtemos as seguintes tabelas:

Note que obteremos três tabelas distintas que surgirão após da normalização 2FN:

Código Curso	Código Aluno	Data de inscrição
0123	01211131	03/05/2011
0222	03121457	02/05/2011
0311	03210001	25/04/2011
0311	15663011	30/04/2011
0425	01211131	04/03/2011

Figura 12.6 Tabela Inscrição-Vestibular

Código Aluno	Nome do Aluno
01211131	Lucas Santos
03121457	Maria Silva
03210001	Jonas Pereira
15663011	Sônia Castro
01211131	Lucas Santos

Figura 12.7 Tabela Aluno-Vestibular

Código Curso	Número da sala
0123	101
0222	103
0311	101
0311	105
0425	110

Figura 12.8 Tabela Curso-Sala

## Início da atividade

### Atividade 3 - Atende ao objetivo 3

A seguir normalize a tabela Pedido-Eletrodoméstico na 2FN. Nesta tabela temos diversos eletrodomésticos e seus correspondentes números de pedidos, data de emissão, nome e identidade da empresa fornecedora, endereço, código, nome, quantidade e preço unitário de cada produto.

Tabela Pedido-Eletrodoméstico

Código Pedido	Código Produto	Data Emissão Pedido	Nome Empresa Fornecedora	CNPJ	Rua	Número	Estado	Nome Produto	Qtde	Preço Unit.
11	191	11/02/11	Casas Ricardo e Ana	02314418992	Mira	11	SP	Fogão	6	450,99
11	022	11/02/11	Casas Ricardo e Ana	02314418992	Flor	231	RJ	Ferro	30	32,00
13	083	20/03/11	Ponto Quente	03210062121	Fé	21	RJ	Lavadora	10	890,99
14	111	07/04/11	Loja Paulão	02121212123	Kan	565	SP	Liquidificador	21	79,99
14	156	07/04/11	Loja Paulão	02121212123	Lua	1290	SP	Geladeira	12	1200
15	112	09/05/11	Ponto Azul	09121212122	Ana	343	SC	Batedora	30	89,99
16	078	09/05/11	Casa Bahia123	01411981000	Fogo	103	AL	TV	20	2100

### Resposta Comentada

Aplicando-se a regra 2FN sobre a tabela Pedido-Eletrodoméstico, deveremos:

- A tabela já está na 1FN.
- A chave primária estará composta pelos atributos código do pedido e o código do produto.
- Analisar os atributos não chaves com dependência na chave primária:
  - Independente do código produto, conhecendo o código do pedido, podemos determinar a data emissão, nome da empresa fornecedora, CNPJ, rua, número e estado. Cada pedido diferente esses atributos podem ser determinados.
  - Conhecendo o código do produto podemos determinar o nome do produto independente ao código do pedido.
  - Para se conhecer as quantidades que foram pedidas à empresa fornecedora é necessário saber qual foi o código do pedido e o código do produto. Considerando as informações do código do pedido teremos várias quantidades, uma para cada produto desse pedido. Se consideramos o

código do produto teremos várias quantidades, uma para cada pedido onde apareça esse produto. Neste caso mostra uma dependência total.

→ Se o atributo preço unitário for associado ao produto, esse atributo possuirá dependência parcial para a chave código de produto.

Criando novas entidades com os atributos dependentes parcialmente das chaves e excluindo esses atributos na tabela original Pedido-Eletrrodoméstico, obteremos as seguintes tabelas:

### 1. Atributos dependentes totalmente da chave

Código Pedido	Código Produto	Qtde	Preço Unit.
11	191	6	450,99
11	022	30	32,00
13	083	10	890,99
14	111	21	79,99
14	156	12	1200
15	112	30	89,99
16	078	20	2100

### 2. Atributos dependentes parcialmente da chave código de pedido:

Código Pedido	Data Emissão Pedido	Nome Empresa Fornecedora	CNPJ	Rua	Número	Estado
11	11/02/11	Casas Ricardo e Ana	02314418992	Mira	11	SP
11	11/02/11	Casas Ricardo e Ana	02314418992	Flor	231	RJ
13	20/03/11	Ponto Quente	03210062121	Fé	21	RJ
14	07/04/11	Loja Paulão	02121212123	Kan	565	SP
14	07/04/11	Loja Paulão	02121212123	Lua	1290	SP
15	09/05/11	Ponto Azul	09121212122	Ana	343	SC
16	09/05/11	Casa Bahia123	01411981000	Fogo	103	AL

### 3. Atributo nome de produto dependente parcialmente da chave código de produto.

Código Produto	Nome Produto
191	Fogão
022	Ferro
083	Lavadora
111	Liquidificador
156	Geladeira

112	Batedora
078	TV

Assim, a tabela Pedido-Eletrdoméstico foi normalizada 2FN, obtendo três tabelas.

### Fim da atividade

O processo de normalização da 3FN também dará origem a novas tabelas; este processo garante uma maior fidelidade ao processo de normalização, completando a 2FN.

### Terceira Forma Normal (3FN)

A forma 3FN estende a forma 2FN para incluir a eliminação de dependências transitivas. A Dependência Transitiva surge quando no momento em que um atributo está dependendo de outro atributo que depende da chave primária. Exemplo:

*Atributo A1 depende do Atributo B1;*

*Atributo B1 depende do Atributo C1 compondo a chave primária;*

*Logo, existe uma dependência transitiva do Atributo A1 em relação ao Atributo C1.*

*Regra 3FN: A tabela deve estar na 2FN e se nenhum atributo não pertencente à chave fica determinada transitivamente por esta.*

Vejamos:

- Para que se aplique a regra da 3FN deve garantir os passos anteriores (2FN e 1FN). Sem eles não haverá sentido aplicar a 3FN.
- “...se nenhum atributo não pertencente à chave...”, este texto sugere que deverão ser analisados os atributos não pertencentes à chave.
- ““...fica determinada transitivamente por esta...”, a dependência transitiva de uma chave, só será possível se a tabela tiver ao menos duas colunas (ou atributos) não pertencentes à chave. Se a tabela possui uma só coluna que não pertence à chave primária, então essa tabela já está na 3FN.
- Para os atributos dependentes transitivamente da chave:
  - Criar novas entidades onde a chave primária será(ão) a(s) coluna(s) que determinou(aram) o valor da coluna analisada. Agregar a essas entidades as colunas dependentes transitivamente.
  - Excluir da entidade original as colunas dependentes transitivamente das chaves mantendo, porém, a coluna determinante da transitividade na tabela.

Exemplo:

Imagine que temos a tabela de compras dos clientes de uma loja de um Shopping e a tabela contendo os nomes dos clientes das lojas.

<b>Código Cliente</b>	Preço Unitário	Qtes itens	Data última compra	Codigo Cidade	Tipo Cidade
03413	31,99	14	01/12/2009	17	interior
07342	21,99	15	04/03/2011	22	capital
01212	47,99	20	12/05/2011	13	interior
04538	99,99	12	1/04/2011	44	metrópole
07342	12,99	28	1/04/2011	15	capital

Figura 12.9 Tabela Compra-Cliente

<b>Código Cliente</b>	Nome Cliente
03413	Ana Pereira
01212	Karla Costa
04538	João Santos
07342	Maria Hagin

Figura 12. 10 Tabela Cliente

- A Tabela Compra-Cliente, os atributos preço unitário (descontos especiais por clientes), quantidades de itens comprados, data da última compra, município e o tipo de cidade onde reside possuem dependência total com a chave primária (código de cliente). Todo cliente é avaliado em função desses atributos.
- O tipo de cidade pode ser determinado através do nome da cidade. O nome da cidade é determinado pelo código do cliente (chave primária), então existe uma dependência transitiva para o atributo tipo de cidade em relação ao atributo código do cliente.
- Para o atributo tipo de cidade, que é dependente transitivamente da chave, devemos criar uma nova entidade em que a chave primária será o atributo que determinou o valor do atributo analisado (código da cidade), agregando a essa entidade o atributo tipo de cidade.
- Devemos excluir da tabela origem (Compra-Cliente) o atributo tipo de cidade mantendo o atributo (nome da cidade) determinante na transitividade na tabela.
- A Tabela Clientes, possui uma só coluna que não pertence à chave primaria; então essa tabela já está na 3FN.

Aplicando a regra 3FN, obtemos as seguintes tabelas:

<b>Código Cliente</b>	Preço Unitário	Qtes itens	Data última compra	Codigo Cidade
03413	31,99	14	01/12/2009	17
07342	21,99	15	04/03/2011	22
01212	47,99	20	12/05/2011	13
04538	99,99	12	1/04/2011	44
07342	12,99	28	1/04/2011	15

Figura 12. 11 Tabela Compra-Cliente Modificada

Codigo Cidade	Tipo Cidade
17	interior
22	capital
13	interior
44	metrópole
15	capital

Figura 12. 12 Tabela Cidade Criada

<b>Código Cliente</b>	Nome Cliente
03413	Ana Pereira
01212	Karla Costa
04538	João Santos
07342	Maria Hagin

Figura 12. 13 Tabela Cliente Mantida

## Início da atividade

### Atividade 4 - Atende ao objetivo 4

A seguir normalize a tabela Produção na 3FN.

Código	Nome	Tempo Produção(horas)	Código Chefe	Nome Chefe
0123	Matéria 1	100	0017	Carlos
3431	Matéria 2	150	0459	Pedro
5486	Matéria 3	80	0321	Augusto
3421	Matéria 4	220	9651	Ana

### Resposta Comentada

- A Tabela Produção, os atributos código, nome, tempo de produção, código do chefe e o nome do chefe possuem dependência total com a chave primária código da matéria prima.
- O nome do chefe pode ser determinado através do código do mesmo. O código do chefe é determinado pelo código da matéria-prima, então existe uma dependência transitiva para o atributo nome do chefe em relação ao atributo código da matéria prima.
- Para o atributo nome do chefe, que é dependente transitivamente da chave, devemos criar uma nova entidade em que a chave primária será o atributo que determinou o valor do atributo analisado (código do chefe), agregando a essa entidade o atributo nome do chefe.

**Aplicando a regra 3FN, obtemos as seguintes tabelas:**

Código	Nome	Tempo Produção(horas)	Código Chefe
0123	Matéria 1	100	0017
3431	Matéria 2	150	0459
5486	Matéria 3	80	0321
3421	Matéria 4	220	9651

e

Código Chefe	Nome Chefe
0017	Carlos
0459	Pedro
0321	Augusto
9651	Ana

## Fim da atividade

## **Conclusão**

Todo banco de dados relacional deve ser projetado para atingir requisitos de qualidade, desempenho e escalabilidade dos dados. A eficiência devem ser mantidos num estado consistente e lógico. A normalização apoia a definição de requisitos de projeto de modelo de dados removendo as falhas de consistências de dados e anomalias de manipulação de dados, preservando os elementos de um sistema de alto desempenho.

## **Resumo**

Normalização:

- É o processo formal passo a passo que examina os atributos de uma entidade, com o objetivo de evitar anomalias observadas na inclusão, exclusão e alteração de registros.
- Normalização de Dados consiste em definir o formato lógico adequado para as estruturas de dados identificados no projeto lógico do sistema, com o objetivo de minimizar o espaço utilizado pelos dados e garantir a integridade e confiabilidade das informações.

Benefícios:

- Estabilidade do Modelo Lógico: capacidade de um modelo manter-se inalterado face a mudanças que venham acontecer.
- Flexibilidade: aumenta o grau de adaptabilidade do banco de dados em função de mudanças no Mundo Real (Regras de Negócio e Requisitos de Informação)
- Integridade: elimina as redundâncias dos dados na tabela.
- Economia: minimiza os custos de qualquer esforço, tempo e armazenamento.
- Expressividade: aumenta o grau de consistência das informações armazenadas no banco de dados.

## **Formas Normais**

- A normalização é feita através da análise dos dados que compõem as estruturas utilizando o conceito chamado “Formas Normais (FN)”. As FN são conjuntos de restrições nas quais os dados devem satisfazê-las.
- A normalização completa dos dados é feita seguindo as restrições das três formas normais existentes, sendo que a passagem de uma FN para outra é feita tendo como base o resultado obtido na etapa anterior, ou seja, na FN anterior.



## **Primeira Forma Normal (1FN)**

*Regra 1FN: Cada atributo de uma tabela deve ter apenas um único valor.*

Devemos,

1. transformar os atributos em atributos atômicos;
2. extrair com os grupos de dados repetidos.

## **Segunda Forma Normal (2FN)**

*Regra 2FN: Além de atingir a 1NF, todos os atributos que não são chaves não dependem parcialmente dessa chave.*

## **Terceira Forma Normal (3FN)**

A forma 3FN estende a forma 2FN para incluir a eliminação de dependências transitivas. A Dependência Transitiva surge quando no momento em que um atributo está dependendo de outro atributo que depende da chave primária. Exemplo:

*Atributo A1 depende do Atributo B1;*

*Atributo B1 depende do Atributo C1 compondo a chave primária;*

*Logo, existe uma dependência transitiva do Atributo A1 em relação ao Atributo C1.*

*Regra 3FN: A tabela deve estar na 2FN e se nenhum atributo não pertencente à chave fica determinada transitivamente por esta.*

## **Referências Bibliográficas**

CARVALHO C.R., 2006. SQL-Guia Prático. 2ed. Rio de Janeiro. Brasport.

DATE C.J., 2003. Introdução a Sistemas de Bancos de Dados. 8ed. Americana. Rio de Janeiro. Elsevier.

ELMASRI R., Navathe S., 2009. Sistemas de Banco de Dados. 5ed. São Paulo. Pearson Addison Wesley.

HEUSER C.A. 2009. Projeto de Banco de Dados. 4ed. Porto Alegre. Bookman.

SETZER V.W. & Corrêa da Silva F.S. 2005. Bancos de Dados. 1ed. São Paulo. Edgard Blucher.

SILBERSCHATZ A., Korth H., 2008. Sistema de Banco de Dados. 3ed. São Paulo. Pearson.