



Banco de dados

Autor : Marcelo Passos dos Santos





Objetivo

Normalização.



Normalização

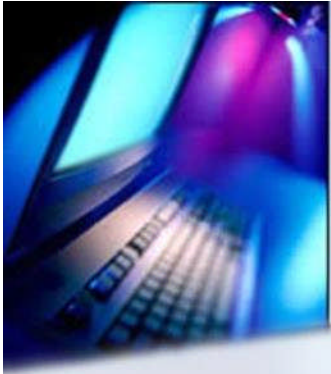
- O objetivo da normalização de um banco de dados é evitar os problemas que podem provocar falhas no projeto do banco de dados, bem como eliminar a mistura de assuntos e as correspondentes redundâncias dos dados desnecessários.
- A abreviação usada é (*NF*), vem do inglês, "Normal Form".



Normalização

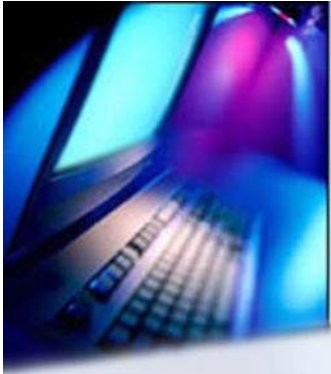
- Normalização

- O processo de normalização aplica uma série de regras sobre as entidades de um banco de dados, para verificar se estão corretamente projetadas.
- Existem diversas formas normais. Entre as mais conhecidas, temos: a primeira forma normal, a segunda forma normal e a terceira forma normal.



Formas Normais - 1FN

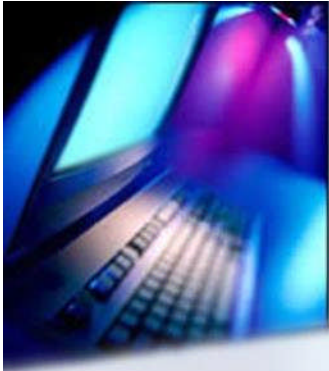
- Uma entidade está na 1FN se todos os seus atributos forem atômicos (simples), ou seja, não são permitidos **atributos multivalorados ou atributos compostos**.



Formas Normais - 1FN

■ Procedimentos :

- a) Identificar a chave primária da entidade;
- b) Identificar o grupo repetitivo e removê-lo da entidade;
- c) Criar uma nova entidade com a chave primária da entidade anterior e o grupo repetitivo. **A chave primária da nova entidade será obtida pela concatenação da chave primária da entidade inicial e a do grupo repetitivo.**
- d) Identificar o atributo multivalorado e decompor.



Formas Normais - 1FN

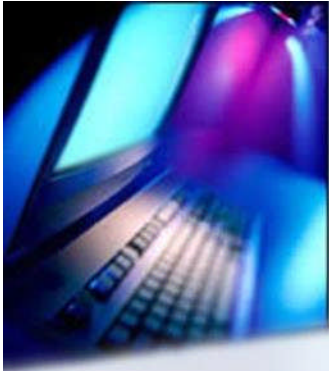
- Exemplo :

- FUNCIONÁRIOS = {CODFUNC + NOME + CARGO + {PROJETO + DATAINI + DATA FIM}}

- Para colocar na 1FN:

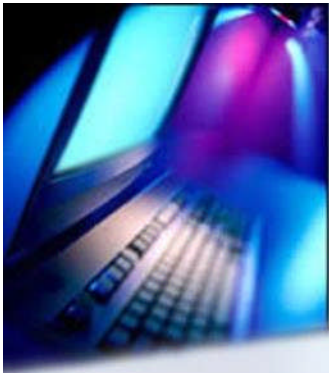
- FUNCIONÁRIOS = {CODFUNC + NOME + CARGO}

- FUNC_PROJ = {CODFUNC + PROJETO + DATAINI + DATA FIM}



Formas Normais - 1FN

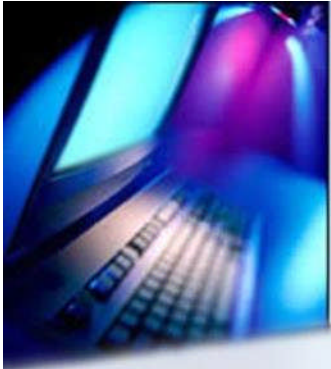
- Exercício : Normalize a tabela abaixo :
- CLIENTE = { Código_cliente + Nome + * Telefone + Endereço }



Formas Normais - 1FN

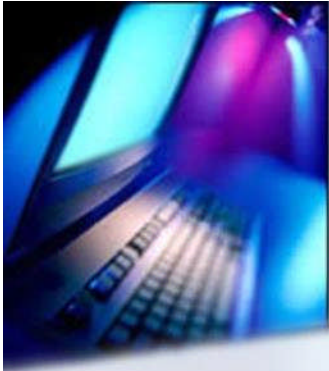
- Resposta : Analisando teremos:
- Todos os clientes possuem Rua, CEP e Bairro, e essas informações estão no mesmo atributo composto (**endereço**), logo ela não está na **primeira forma normal**.
- Para normalizar, deveremos colocar cada informação em um atributo diferente, como no exemplo a seguir:

CLIENTE = {Código_cliente + Nome + * Telefone + Logradouro + Bairro +
Complemento + Cidade + Uf + CEP }



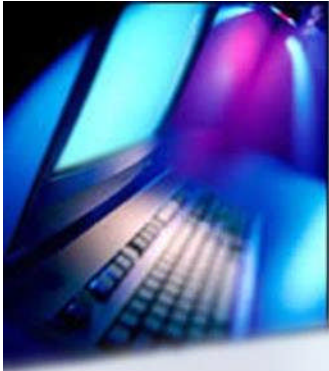
Formas Normais - 1FN

- Resposta : Analisando teremos:
- Mesmo com o ajuste acima, a tabela ainda não está na **primeira forma normal**, pois há clientes com mais de um telefone e os valores estão em uma mesmo atributo.
- Para normalizar será necessário criar uma nova tabela para armazenar os números dos telefones e o campo-chave da tabela cliente. Veja o resultado a seguir:
- **CLIENTE** = {Código cliente + Nome + Logradouro + Bairro + Cidade + Uf + CEP }
- **TELEFONE** = {Código cliente + Telefone }



Formas Normais - 2FN

- Uma tabela está na **Segunda Forma Normal** 2FN se ela estiver na 1FN e todos os atributos **não chave forem totalmente dependentes da chave primária** (dependente de toda a chave e não apenas de parte dela).
- Em outras palavras, se algum campo depender somente de **parte** de uma chave composta (e não da chave composta como um todo), então este campo deve ser extraído para outra tabela.



Formas Normais - 2FN

- Procedimentos :
 - a) Identificar os atributos que não são funcionalmente dependentes de toda a chave primária;
 - b) Remover da entidade todos esses atributos identificados e criar uma nova entidade com eles.



Formas Normais - 2FN

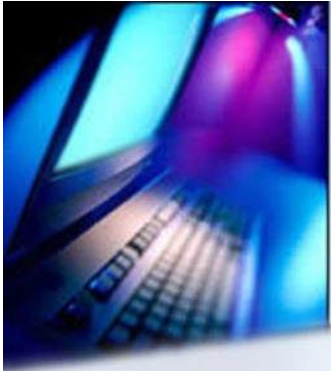
■ Exemplo :

- $ALUNOS_CURSOS = \{\underline{ID_ALUNO} + \underline{ID_CURSO} + NOTA + DESCRICAO_CURSO\}$
- O problema consiste no fato de que o campo DESCRICAO_CURSO depende somente do campo ID_CURSO (apesar da chave ser composta), que é parte da chave primária da tabela.
- Para que esta tabela esteja na 2NF, o campo DESCRICAO_CURSO deve ser extraído para outra tabela:
- $ALUNOS_CURSOS = \{\underline{ID_ALUNO} + \underline{ID_CURSO} + NOTA\}$
- $CURSOS = \{\underline{ID_CURSO} + DESCRICAO\}$



Formas Normais - 2FN

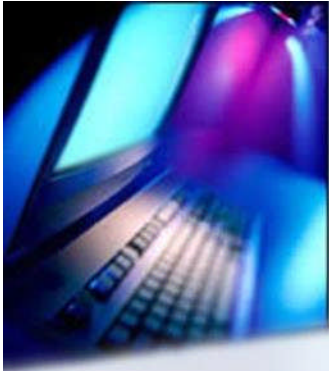
- Exercício : Normalize a tabela abaixo :
- VENDAS = { N_pedido + Código_produto + Produto + Quant + Valor_unit + Subtotal }



Formas Normais - 2FN

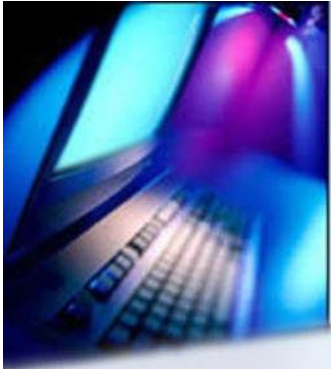
- Resposta : Analisando teremos:

- O nome do produto depende do código do produto, porém não depende de N_pedido que é a chave primária da tabela, portanto não está na **segunda forma normal**.
- Isto gera problemas com a manutenção dos dados, pois se houver alteração no nome do produto teremos que alterar em todos os registros da tabela venda.
- Para normalizar esta tabela teremos de criar a tabela Produto que ficará com os atributos Código_produto e produto e na tabela Venda manteremos somente os atributos N_pedido, código_produto, quant, valor_unit e subtotal. Veja o resultado abaixo:
- **VENDAS = { N_pedido + Código_produto + Quant + Valor_unit + Subtotal }**
- **PRODUTO = { Código_produto + Produto }**



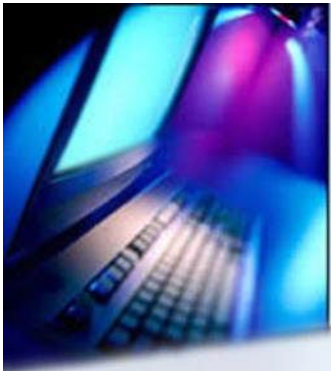
Formas Normais - 3FN

- Uma tabela está na **Terceira Forma Normal** 3FN se ela estiver na 2FN e se nenhuma coluna não-chave depender de outra coluna não-chave.
- Em outras palavras, caso exista um ou mais atributos que dependam de um atributo não-chave, temos de eliminar aqueles campos que podem ser obtidos pela equação de outros campos da mesma tabela.



Formas Normais - 3FN

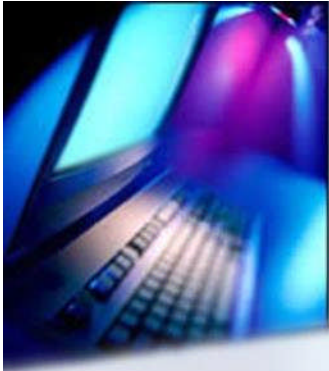
- Procedimentos :
 - a) Identificar todos os atributos que são funcionalmente dependentes de outros atributos não chave;
 - b) Removê-los.



Formas Normais - 3FN

■ Exemplo :

- $\text{FUNCIONARIOS} = \{\underline{\text{ID}} + \text{NOME} + \text{ID_CARGO} + \text{DESCRICAO_CARGO}\}.$
- O problema é que o campo DESCRICAO_CARGO depende exclusivamente do campo ID_CARGO, que não faz parte da chave da tabela.
- Para deixar esta tabela na 3NF, basta extrair o campo para outra tabela:
- $\text{FUNCIONARIOS} = \{\underline{\text{ID}} + \text{NOME} + \text{ID_CARGO}\}$
- $\text{CARGOS} = \{\underline{\text{ID_CARGO}} + \text{DESCRICAO}\}$
- **Observação importante:** Muitas pessoas confundem a 2NF com a 3NF. Entretanto, a 2NF proíbe que determinados atributos dependam de **parte de uma chave composta**, enquanto que a 3NF proíbe que determinados atributos dependam de **atributos não-chave**



Formas Normais - 3FN

- Exercício : Normalize a tabela abaixo :
- VENDAS = { N_pedido + Código_produto + Produto + Quant + Valor_unit + Subtotal }



Formas Normais - 3FN

- Resposta : Analisando teremos:
 - Considerando a tabela Venda, veremos que a mesma não está na **terceira forma normal**, pois o subtotal é o resultado da multiplicação Quant X Valor_unit, desta forma a coluna subtotal depende de outras colunas não-chave.
 - Para normalizar esta tabela na **terceira forma normal** teremos de eliminar a coluna subtotal, como no exemplo a seguir:

VENDAS = { N_pedido + Código produto + Quant + Valor_unit }

PRODUTO = { Código produto + Produto }