

Definição: é um processo para avaliar e corrigir estruturas e tabelas de modo a minimizar as redundâncias de dados e colaborando com a redução das anomalias.

A normalização consiste, em princípio, de 3 fases:

- 1ª. Forma Normal – 1FN
- 2ª. Forma Normal – 2FN
- 3ª. Forma Normal – 3FN

Do ponto de vista estrutural a 3FN é melhor do que a 2FN, que é melhor do que a 1FN.

Vale destacar que quanto mais elevada for a FN maior será a quantidade de relacionamentos.

Visando melhor desempenho, permite-se a desnormalização de certas partes do projeto, sob pena da ocorrência de redundâncias.

O objetivo da normalização é garantir que todas as tabelas atendam ao conceito de relações bem estabelecidas, ou seja, que tenham as características abaixo:

- Cada tabela representa um único assunto.
Ex.: a tabela disciplina conterá apenas os dados diretamente relacionados as disciplinas.

-Nenhum registro será armazenado desnecessariamente em mais de uma tabela. Objetivando que os dados sejam atualizados em apenas um lugar.

-Todos os dados não primários da tabela são dependentes da PK para garantir a unicidade.

Considere um atributo A que numa dada relação R determina a unicidade de outro atributo B em qualquer momento da vida do BD. Considere que B será funcionalmente dependente de A .

Diz-se então que:

A determina funcionalmente B e B é determinado funcionalmente por A .

$$A \rightarrow B$$

Exercícios

Dada a seguinte relação:

cliente (nro_cliente, nome, endereço)

Verifique se as seguintes dependências são corretas:

nro_cliente \rightarrow nome

nro_cliente \rightarrow endereço

nome \rightarrow endereço

Exercícios

Dada a seguinte relação:

cliente (nro_cliente, nome, endereço)

Verifique se as seguintes dependências são corretas:

nro_cliente \rightarrow nome

nro_cliente \rightarrow endereço

nome \rightarrow endereço

Dependência Funcional

Dada a seguinte relação:

linha_pedido (nro_pedido, nro_peça, qtde_comprada, preço_cotado,)

E a seguinte tabela:

Nro_ped	Nro_peça	Qtde_comp	Preç_cotado
101	80	10	52
101	85	02	22
102	85	08	21

- As seguintes dependências são verdadeiras?

Nro_pedido --> qtde_comprada

Nro_peça --> qtde_comprada

Nro_pedido --> preço_cotado

Nro_peça --> preço_cotado

Nro_pedido, Nro_peça --> qtde_comprada

Nro_pedido, Nro_peça --> preço_cotado

PRIMEIRA FORMA NORMAL (1FN)

- Geralmente considera parte da definição formal de uma relação.
- Não permite atributos multivalorados, compostos ou suas combinações.
- Não permite conjunto de valores, tupla de valores ou uma combinação de ambos.

Uma boa modelagem ER geralmente origina relações que já se encontram na 1FN

PRIMEIRA FORMA NORMAL (1FN)

Uma relação está na 1FN se:

- todo valor em R for atômico (ou seja, R não contém grupos de repetição)

Funcionário

ID_func	Nome	Filhos	Nasc_filhos
1001	Jane	Maria, Jonas	1/1/1992, 5/10/1995
1002	Samanta	Paulo, Pedro, Augusto	20/4/1990, 6/8/2000, 2/5/2002
1003	Alberto	Vanessa, Marcela	20/10/1991, 5/6/1993

Problemas com o exemplo apresentado:

- Não há como saber qual exatamente é a data de nascimento de cada criança, nada garante a seqüência que é mostrada;
- A pesquisa na tabela fica difícil, pois o domínio sugere caracteres, pois não se sabe quantos são os filhos e respectivas datas de nascimento.

Solução incorreta:

Funcionário

ID_func	Nome	Filho1	Nas_filh1	Filho2	Nas_filh2	Filho3	Nas_filh3
1001	Jane	Maria	1/1/1992	Jonas	5/10/1995		
1002	Samanta	Paulo	20/4/1990	Pedro	16/8/2000	Augusto	26/5/2002
1003	Alberto	Vanessa	20/10/1991	Marcela	15/6/1993		

Problemas com a solução:

A relação está limitada a três filhos por funcionário;

Pode-se ter muitos funcionários com apenas um filho ou sem filhos;

Pesquisar por um filho específico aumenta a complexidade da consulta.

Solução correta :

Funcionário

ID_func	Nome
1001	Jane
1002	Samanta
1003	Alberto

Filhos

ID_func	Nome_filho	Nascimento
1001	Maria	1/1/1992
1001	Jonas	5/10/1995
1002	Paulo	20/4/1990
1002	Pedro	16/8/2000
1002	Augusto	26/5/2002
1003	Vanessa	20/10/1991
1003	Marcela	15/6/1993

- As tabelas não apresentam problema com repetição;
- Não há limite para o número de filhos por funcionário;
- Não há desperdício de espaço;
- Realizar uma pesquisa requer uma consulta simples por um único atributo.

Exercício

Dada a seguinte relação, convertê-la para a 1FN:

pedido (nro_pedido, data, número_cliente,
nome_cliente, {nro_peça, nome_peça,
qtde_comprada, preço_cotado})

1FN:

pedido (nro_pedido, data, número_cliente,
nome_cliente, nro_peça, nome_peça, qtde_comprada,
preço_cotado)

1 FN

Para que serve a 1FN:

- para evitar que se tenha que reservar espaços para armazenar dados múltiplos, sendo que o espaço pode ser desperdiçado em um registro e ser insuficiente em outro (quantos itens tem uma nota fiscal?).

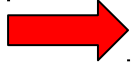
Como se usa a 1FN:

- projetam-se os atributos com domínio multivalorado para fora da tabela, levando um atributo (geralmente a chave da tabela original) como elo para refazer a ligação e recuperar o conteúdo da tabela original.

SEGUNDA FORMA NORMAL (2FN)

Uma relação está na 2FN se:

- está na 1FN
- não existe atributo não chave que é dependente de somente uma parte da chave primária.



não existir atributo não chave que seja dependente de somente uma parte da chave primária.

Exemplo:

pedido (nro_pedido, data, número_cliente, nome_cliente, nro_peça, nome_peça, qtde_comprada, preço_cotado)

nro_ped \rightarrow data, número_cliente, nome_cliente

nro_peça \rightarrow nome_peça

nro_ped, nro_peça \rightarrow qtde_comprada, preço_cotado

Método para corrigir o problema:

- para cada subconjunto do conjunto de atributos que constitui a chave primária, começar uma relação com esse subconjunto como sua chave primária;
- incluir os atributos da relação original na relação correspondente à chave primária apropriada, isto é, colocar cada atributo junto com a coleção mínima da qual ele depende, atribuindo um nome a cada relação.

Problema:

pedido (nro_pedido, data, número_cliente, nome_cliente, nro_peça, nome_peça, qtde_comprada, preço_cotado)

Solução:

pedido (nro_ped, data, número_cliente, nome_cliente)

peça (nro_peça, nome_peça)

pedido_peça (nro_ped, nro_peça, qtde_comprada, preço_cotado)

TERCEIRA FORMA NORMAL (3FN)

Uma relação r está na 3FN se:

- está na 2FN
- não existem atributos não chave que sejam dependentes de outros atributos não chave

pedido (nro_ped, data, número_cliente, nome_cliente)

peça (nro_peça, nome_peça)

pedido_peça (nro_ped, nro_peça, qtde_comprada,
preço_cotado)

Método para corrigir o problema:

- para cada determinante que não é uma chave candidata, remover da relação os atributos que dependem desse determinante;
- criar uma nova relação contendo todos os atributos da relação original que dependem desse determinante;
- tornar o determinante a chave primária da nova relação.

Corrigindo o problema

pedido (nro_ped, data, **número_cliente**)

peça (nro_peça, nome_peça)

pedido_peça (nro_ped, nro_peça,
qtde_comprada, preço_cotado)

cliente (**número_cliente**, nome_cliente)

Exercícios:

Normalizar a seguinte relação

aluno (nro_aluno, cod_depto, nome_depto, sigla_depto, orient, nome_orient, fone_orient, nro_créd, cod_curso)

- Está na 1FN pois não possui grupos de repetição
- Está na 2FN pois não existem atributos não chave que são dependentes de parte da chave primária
- A relação não está na 3FN pois existem atributos não chave que são dependentes de outros atributos não chave

nro_aluno → cod_orient, nro_créd, cod_curso

cod_depto → nome_depto, sigla

cod_orient → nome_orient, fone_orient, cod_depto

aluno (nro_aluno, cod_orient, nro_créd, cod_curso)

departamento (cod_depto, nome_depto, sigla)

orientador (cod_orient, nome_orient, fone_orient, cod_depto)

Exercícios: Normalizar a seguinte relação

aluno (nro_aluno, nome_aluno, nro_créditos, nro_depto, nome_depto, {nro_curso, descrição_curso, ano_ingresso})

*** 1 aluno pode estar em mais de um curso*

- 1FN

aluno (nro_aluno, nome_aluno, nro_créditos, nro_depto, nome_depto, nro_curso, descrição_curso, ano_ingresso)

- 2FN

nro_aluno → nome_aluno

nro_curso → nro_depto, nome_depto, descrição_curso

nro_aluno, nro_curso → nro_créditos, ano_ingresso

aluno (nro_aluno, nome_aluno)

curso (nro_curso, nro_depto, nome_depto, descrição_curso)

curso (nro_aluno, nro_curso, nro_créditos, ano_ingresso)

- 3FN

aluno (nro_aluno, nome_aluno)

curso (nro_aluno, nro_curso, nro_créditos, ano_ingresso)

curso (nro_curso, nro_depto, descrição_curso)

departamento (nro_depto, nome_depto)

Exercícios: Normalizar a seguinte relação

paciente (nro_paciente, nome_paciente, nro_quarto, descrição_quarto, nro_cômodos_quarto, {cod_médico, nome_médico, fone_médico})

- 1FN

paciente (nro_paciente, nome_paciente, nro_quarto, descrição_quarto, nro_cômodos_quarto, cod_médico, nome_médico, fone_médico)

- 2FN

nro_paciente → nome_paciente, nro_quarto, descrição_quarto, nro_cômodos_quarto

cod_médico → nome_médico, fone_médico

nro_paciente, cod_médico →

paciente (nro_paciente, nome_paciente, nro_quarto, descrição_quarto, nro_cômodos_quarto)

médico (cod_médico, nome_médico, fone_médico)

atende (nro_paciente, cod_médico)

- 3FN

nro_paciente → nome_paciente, nro_quarto

nro_quarto → descrição_quarto, nro_cômodos_quarto

paciente (nro_paciente, nome_paciente, nro_quarto)

médico (cod_médico, nome_médico, fone_médico)

atende (nro_paciente, cod_médico)

quarto (nro_quarto, descrição_quarto, nro_cômodos_quarto)