

Aula 8: Normalização de Base de Dados Relacional

Introdução

A **normalização de dados** é uma série de passos que se seguem no projeto de um banco de dados, que permitem um armazenamento consistente e um eficiente acesso aos dados em bancos de dados relacionais. Esses passos reduzem a redundância de dados e as chances dos dados se tornarem inconsistentes.

No entanto, muitos SGBDs relacionais não têm separação suficiente entre o projeto lógico da base de dados e a implementação física do banco de dados, e isso tem como consequência que as consultas feitas a um banco de dados totalmente normalizado tenham mau desempenho. Nestes casos, usa-se por vezes a desnormalização para melhorar o desempenho, com o custo de menores garantias de consistência.

Paranômica Informal

Diz-se que uma tabela num banco de dados relacional está numa certa forma normal se satisfaz certas condições. O trabalho original de Edgar F. Codd definiu três dessas formas, mas existem hoje outras formas normais geralmente aceites. Damos aqui uma curta panorâmica informal das mais comuns. Cada forma normal listada abaixo representa uma condição mais forte que a precede na lista. Para a maioria dos efeitos práticos, considera-se que as bases de dados estão normalizadas se aderirem à terceira forma normal.

Primeira Forma Normal (ou 1FN) requer que todos os valores de colunas em uma tabela sejam atômicos (exemplo: um número é um átomo, enquanto uma lista ou um conjunto não o são). A normalização para a primeira forma normal elimina grupos repetidos, pondo-os cada um em uma tabela separada, conectando-os com uma chave primária ou estrangeira;

Segunda Forma Normal (ou 2FN) requer que não haja dependência funcional não-trivial de um atributo que não seja a chave, em parte da chave candidata;

Terceira Forma Normal (ou 3FN) requer não haver dependências funcionais não-triviais de atributos que não sejam chave, em qualquer coisa exceto um superconjunto de uma chave candidata;

Primeira Forma Normal

Uma relação estará na primeira forma normal 1FN, se não houver grupo de dados repetidos, isto é, se todos os valores forem únicos. Em outras palavras podemos definir que a primeira forma normal não admite repetições ou campos que tenha mais que um valor.

Os procedimentos mais recomendados para aplicar a 1FN são os seguintes:

- Identificar a chave primária da entidade;
- Identificar o grupo repetitivo e removê-lo da entidade;
- 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.

Exemplo

Considere a tabela cliente abaixo:

Código_cliente	Nome	Telefone	Endereço
C001	José	9563-6352 9847-2501	Rua Seis, 85 Morumbi 12536-965
C002	Maria	3265-8596	Rua Onze, 64 Moema 65985-963
C003	Janio	8545-8956 9598-6301	Praça ramos Liberdade 68858-633

Tabela 1: Tabela desnormalizada, ou seja, não está na 1ª forma normal.

Todos os clientes possuem Rua, CEP e Bairro, e essas informações estão na mesma célula da tabela, logo ela não está na primeira forma normal. Para normalizar, deveremos colocar cada informação em uma coluna diferente, como no exemplo a seguir:

Código_cliente	Nome	Rua	Bairro	Cep
C001	José	Rua Seis, 85	Morumbi	12536-965
C002	Maria	Rua Onze, 64	Moema	65985-963
C003	Janio	Praça ramos	Liberdade	68858-633

Tabela 2: Tabela ainda não está na primeira forma normal.

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 mesma célula. 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:

Código_cliente	Nome	Rua	Bairro	Cep
C001	José	Rua Seis, 85	Morumbi	12536-965
C002	Maria	Rua Onze, 64	Moema	65985-963
C003	Janio	Praça ramos	Liberdade	68858-633

Tabela 3: Tabela Cliente na primeira forma normal.

Codigo_cliente	Telefone
C001	9563-6352
C001	9847-2501
C002	3265-8596
C003	8545-8956
C003	9598-6301

Tabela 4: Tabela ClienteTelefones na primeira forma normal.

Segunda Forma Normaliza

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).

Se o nome do produto já existe na tabela produtos, então não é necessário que ele exista na tabela de produtos. A segunda forma normal trata destas anomalias e evita que valores fiquem em redundância no banco de dados. Procedimentos:

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

A chave primária da nova entidade será o atributo do qual os atributos do qual os atributos removidos são funcionalmente dependentes.

Exemplo

Considere a tabela vendas abaixo:

N_pedido	Codigo_produto	Produto	Quant	Valor_unit	Subtotal
1005	1-934	Impressora laser	5	1.500,00	7.500,00
1006	1-956	Impressora desjet	3	350,00	1.050,00
1007	1-923	Impressora matricial	1	190,00	190,00
1008	1-908	Impressora mobile	6	980,00	5.880,00

Tabela 5: Tabela não está na segunda forma normal.

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:

Codigo_produto	Produto
1-934	Impressora laser
1-956	Impressora desjet
1-923	Impressora matricial
1-908	Impressora mobile

Tabela 6: Tabela Produto na segunda forma normal.

N_pedido	Codigo_produto	Quant	Valor_unit	Subtotal
1005	1-934	5	1.500,00	7.500,00
1006	1-956	3	350,00	1.050,00
1007	1-923	1	190,00	190,00
1008	1-908	6	980,00	5.880,00

Tabela 7: Tabela Venda na segunda forma normal.

Conforme visto na Primeira forma normal, quando aplicamos normalização é comum gerar novas tabelas a fim de satisfazer as formas normais que estão sendo aplicadas.

Terceira Forma Normaliza

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.

Na terceira forma normal temos de eliminar aqueles campos que podem ser obtidos pela equação de outros campos da mesma tabela. Procedimentos:

- Identificar todos os atributos que são funcionalmente dependentes de outros atributos não chave;
- Removê-los.

A chave primária da nova entidade será o atributo do qual os atributos removidos são funcionalmente dependentes.

Exemplo

Considere a tabela abaixo:

N_pedido	Codigo_produto	Quant	Valor_unit	Subtotal
1005	1-934	5	1.500,00	7.500,00
1006	1-956	3	350,00	1.050,00
1007	1-923	1	190,00	190,00
1008	1-908	6	980,00	5.880,00

Tabela 8: Tabela não está na terceira forma normal.

Considerando ainda a nossa 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:

N_pedido	Codigo_produto	Quant	Valor_unit
1005	1-934	5	1.500,00
1006	1-956	3	350,00
1007	1-923	1	190,00
1008	1-908	6	980,00

Tabela 9: Tabela na terceira forma normal.

Conforme visto na primeira forma normal e segunda forma normal, a normalização torna a tabela mais otimizada e sem anomalias.