



Banco de Dados

An abstract graphic on the left side of the slide, featuring several overlapping green shapes. There is a dark green L-shaped block at the top left, a medium green semi-circular arc in the center, and a light green semi-circular arc at the bottom left. These shapes overlap each other and the white background.

Modelo Lógico

Definições – Modelo Lógico



Compreende uma descrição das estruturas que serão armazenadas no banco e que resulta numa representação gráfica dos dados de uma maneira lógica, inclusive nomeando os componentes e ações que exercem uns sobre os outros.

A técnica de modelagem mais difundida é a abordagem entidade-relacionamento (ER).

Nesta técnica, um modelo conceitual é usualmente representado através de um diagrama, chamado diagrama entidade-relacionamento (DER).

Por enquanto iremos estudar apenas o modelo relacional, no qual os dados estão organizados em forma de tabelas.



Modelo Lógico

Normalização

Normalização é o processo de modelar o banco de dados projetando a forma como as informações serão armazenadas a fim de eliminar, ou pelo menos minimizar, a redundância no banco. Tal procedimento é feito a partir da identificação de uma anomalia em uma relação, decompondo-as em relações melhor estruturadas.

Normalmente precisamos remover uma ou mais colunas da tabela, dependendo da anomalia identificada e criar uma segunda tabela, obviamente com suas próprias chaves primárias e relacionarmos a primeira com a segunda para assim tentarmos evitar a redundância de informações.

Um banco de dados dentro dos padrões de normalização reduz o trabalho de manutenção e ajuda a evitar o desperdício do espaço de armazenamento. Se tivermos cadastrado no banco um cliente e tivermos o seu telefone registrado em mais de uma tabela, havendo uma alteração no seu número de telefone, teremos que fazer essa atualização em cada tabela. A tarefa se torna muito mais eficiente se tivermos seu telefone registrado em apenas uma tabela.

Os próximos parágrafos demonstram melhor as anomalias no banco de dados e as diferentes regras de normalização, bem como a forma de aplicá-las para estruturarmos o banco de dados da melhor maneira possível.

Modelo Lógico

Formas Normais



Como mencionado anteriormente, temos conjuntos de regras para determinar com qual forma normal o banco é compatível. Primeiramente, precisamos verificar se encontramos compatibilidade com a primeira forma normal. Caso esteja tudo conforme, analisamos se a segunda forma normal se encaixa e assim sucessivamente.

É importante lembrar que para uma relação atender as exigências de uma forma normal, se faz necessário que esta obedeça as regras da forma normal anterior. A primeira forma normal é exceção pois não existe uma forma normal anterior a primeira.



Modelo Lógico

Primeira Forma Normal

Uma relação está na primeira forma normal quando todos os atributos contêm apenas um valor correspondente, singular e não existem grupos de atributos repetidos — ou seja, não admite repetições ou campos que tenham mais que um valor.

O procedimento inicial é identificar a chave primária da tabela. Após, devemos reconhecer o grupo repetitivo e removê-lo da entidade. Em seguida, criamos uma nova tabela com a chave primária da tabela anterior e o grupo repetitivo.

| Código | Nome | Endereço | Telefone |
|--------|---------------|--------------------------|--------------------------|
| 1001 | Diego Machado | Rua Tal 321 Porto | 5312345678 5398765432 |
| 1002 | Fulano de Tal | Avenida Tal 71 Centro | 5187654321 5143215678 |

Analisando o exemplo acima, podemos observar dois problemas: temos uma pessoa com dois números de telefone e um endereço com diferentes valores, a rua e o bairro. A fim de normalizar, teremos que colocar cada informação em uma coluna diferente e criar uma nova tabela relacionando a pessoa a seus números de contato.

Modelo Lógico

Primeira Forma Normal



| Código | Nome | Endereço | Bairro |
|--------|---------------|----------------|--------|
| 1001 | Diego Machado | Rua Tal 321 | Porto |
| 1002 | Fulano de Tal | Avenida Tal 71 | Centro |

Dessa forma, como mostrado na tabela acima, temos uma tabela na primeira forma normal evitando assim repetições e campos com múltiplos valores, conforme observamos na tabela abaixo.

| Código | Telefone |
|--------|------------|
| 1001 | 5312345678 |
| 1001 | 5398765432 |
| 1002 | 5112345678 |
| 1002 | 5187654321 |

Modelo Lógico

Segunda Forma Normal

É dito que uma tabela está na segunda forma normal se ela atende a todos os requisitos da primeira forma normal e se os registros na tabela, que não são chaves, dependam da chave primária em sua totalidade e não apenas parte dela. A segunda forma normal trabalha com essas irregularidades e previne que haja redundância no banco de dados.

Para isso, devemos localizar os valores que dependem parcialmente da chave primária e criar tabelas separadas para conjuntos de valores que se aplicam a vários registros e relacionar estas tabelas com uma chave estrangeira.

| cd_locacao | cd_filme | titulo_filme | devolucao | cd_cliente |
|------------|----------|--------------|------------|------------|
| 1010 | 201 | The Matrix | 2011-10-12 | 743 |
| 1011 | 302 | O Grito | 2011-12-10 | 549 |
| 1012 | 201 | The Matrix | 2011-12-30 | 362 |

Podemos observar que a tabela acima apresenta uma coluna responsável por armazenar o título do filme, onde este foi alugado e está associado a um número de locação. Porém, ele também está associado a um código, tornando-o então um valor que não é totalmente dependente da chave primária da tabela.

Modelo Lógico

Segunda Forma Normal



| cd_filme | titulo_filme |
|----------|--------------|
| 201 | The Matrix |
| 302 | O Grito |

Se em algum momento tivermos que alterar o título de um filme, teríamos que procurar e alterar os valores em cada tupla (linha) da tabela. Isso demandaria um trabalho e tempo desnecessário. Porém, ao criarmos uma tabela e vincularmos elas com o recurso da chave estrangeira, tornamos o nosso banco mais organizado e ágil para as futuras consultas e manutenções que podem vir a ser necessárias.

| cd_locacao | cd_filme | devolucao | cd_cliente |
|------------|----------|------------|------------|
| 1010 | 201 | 2011-10-12 | 743 |
| 1011 | 302 | 2011-12-10 | 549 |
| 1012 | 201 | 2011-12-30 | 362 |

Modelo Lógico

Terceira Forma Normal

Se analisarmos uma tupla e não encontrarmos um atributo não chave dependente de outro atributo não chave, podemos dizer que a entidade em questão está na terceira forma normal - contanto que esta não vá de encontro as especificações da primeira e da segunda forma normal.

Como procedimento principal para configurar uma entidade que atenda as regras da terceira forma normal, nós identificamos os campos que não dependem da chave primária e dependem de um outro campo não chave. Após, separamos eles para criar uma outra tabela distinta, se necessário.

| placa | modelo | qtd_kmetro | cod_fab | nome_fab |
|---------|---------|------------|---------|-------------|
| qwel234 | Modelo1 | 867 | 3004 | fabricante1 |
| asd456 | Modelo2 | 928 | 3005 | fabricante2 |

No exemplo acima temos uma entidade que lista os carros cadastrados, bem como o modelo, a quantidade de quilômetros rodados, o código do fabricante e o nome do fabricante. Observamos que “nome_fab” se dá em função de “cod_fab”. Para adequarmos esta tabela de acordo com os padrões da terceira forma normal, devemos remover a coluna do nome do fabricante.



Muito Obrigado!