**NORMALIZAÇÃO**

Material de Apoio

Sumário

￼

￼

4. Por que utilizar?

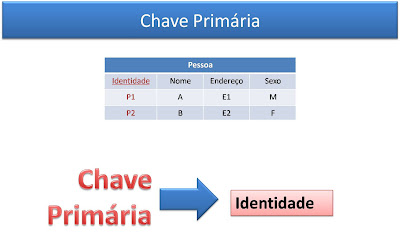
5. Vantagens

6. Formas normais

# **Tipos de Chaves**

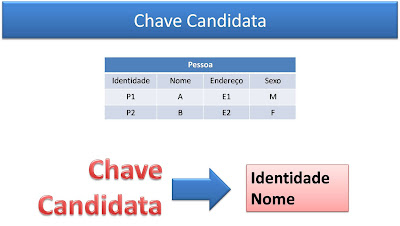
**Chave primaria (PK):**

É a coluna, que recebe a função de identificador de registros da tabela, ou seja, é formada por um único campo, esse campo não pode ter dois ou mais registros de mesmo valor e também não pode conter nenhum registro nulo.



**Chave candidata:**

Ocorrem quando em uma relação existe mais de uma combinação de atributos possuindo a propriedade de identificação única (são as colunas que poderiam ser atribuídas como PK, mas não são).



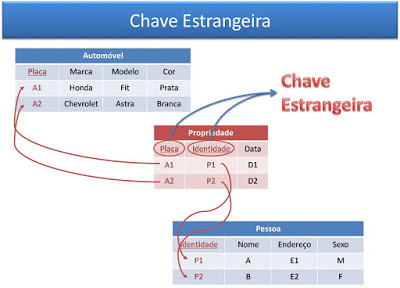
**Superchave:**

Conjunto de colunas que tenha como subconjunto uma chave candidata



**Chave estrangeira (FK):**

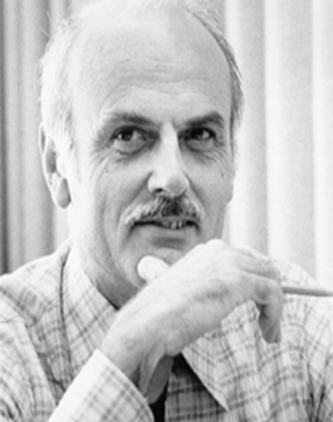
Uma chave da tabela que é PK em outra, criando relação entre as duas, em outras palavras sempre que houver o relacionamento 1:N entre duas tabelas



**Quando Surgiu?**

A normalização surgiu como um processo para se escolher "bons" esquemas de relação, ou seja, para avaliar formalmente porque um conjunto de atributos para um esquema de relação é melhor do que um outro.

Em 1985 E. F. Codd publica um artigo em que definia 13 regras para que um sistema gerenciador de banco de dados fosse considerado relacional, assim, sendo o criador do modelo relacional.



Em 1972 já introduzia o conceito de normalização. No início criou as três primeiras formas normais, 1FN, 2FN e 3FN, com o avanço do tempo uma definição mais forte da terceira norma foi proposta por Boyce-Codd(FNBC) e mais formas foram surgindo.

# **O que é a normalização?**

Normalização é um processo de modelagem de um banco de dados com a finalidade de evitar e diminuir a redundância do banco, projetando a forma como as informações serão armazenadas e eliminar anomalias em registros. Criando assim relações mais bem estruturadas.

Um banco de dados bem estruturado dentro dos padrões da normalização, reduz a dificuldade do trabalho de manutenção, além de reduzir o espaço de armazenamento ocupado. Se os dados estiverem em mais de um local, e precisarem ser alterados, dificulta a manutenção do banco de dados, é muito mais fácil alterar dados que estão localizados em somente 1 local e em nenhum outro lugar

As principais anomalias encontradas em um banco não normalizado são:

**. Inserção**

**. Atualização**

**. Exclusão**

**Por que utilizar?**

Dados redundantes desperdiçam espaço em disco e criam problemas de manutenção. Se os dados existentes em mais de um local devem ser alterados, os dados devem ser alterados exatamente da mesma maneira em todos os locais. Uma alteração de endereço do cliente é muito mais fácil de implementar se esses dados são armazenados apenas na tabela Clientes e em nenhum outro lugar no banco de dados.

Resumindo, a principal razão para usar a normalização em um projeto de banco de dados é exatamente as enormes vantagens que ela traz se comparada a um banco relacional não normalizado.

**Vantagens**

Estabilidade do modelo lógico: capacidade de um modelo manter-se inalterado face às mudanças que venham a ser percebidas ou introduzidas no ambiente que tenha sido modelado.

Flexibilidade: capacidade de adaptação a demandas diferenciadas, a expansão e redução, a omissão ou presença, etc.

Integridade: diz respeito à qualidade do dado. Um dado mapeado em mais de um local de modo diferente, com valores instanciados de modo diferentes, pode ser indício de que não há integridade entre eles.

Economia: no espaço de armazenamento em relação ao custo de manipulação de dados (que representa todo e qualquer esforço, tempo, ou valor agregado ao fato de manipularmos volumes de dados maiores do que os efetivamente necessários); custo causado pelo atraso do fornecimento da informação desejada.

Fidelidade ao ambiente observado: ajuda a definir elementos que foram despercebidos durante o processo de modelagem.

**Formas Normais**

As regras de normalização são chamadas de “formulário normal”, se um banco de dados possui a primeira regra formal, o banco de dados será nomeado de “primeiro formulário normal”, caso seja observado a presença das três primeiras regras formais, o banco de dados será dito como “terceiro formulário formal”

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.

Primeira Forma

Nesta Primeira Forma Normal, os atributos não devem ser repetidos ou multivalorados (com mais de um valor), por exemplo:

|  |  |  |  |
| --- | --- | --- | --- |
|  | IdCliente | Telefone | Endereço |
| José | 001 | 11 98765-4321  11 91234-5678 | Marte |
| Antônio | 002 | 11 95789-9542 | Júpiter |

No exemplo acima, o atributo telefone, possui mais de um valor atribuído, desta forma identificamos a chave primária, depois retiramos o atributo multivalorado e criamos uma outra tabela:

Cliente

|  |  |
| --- | --- |
| Id\_cliente | Endereço |
| 001 | Marte |
| 002 | Júpiter |

Telefone

|  |  |
| --- | --- |
| Id\_Cliente (chave estrangeira) | Telefone |
| 001 | 11 98765-4321  11 91234-5678 |
| 002 | 11 95789-9542 |

Segunda Forma

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

Para estar dentro dos padrões da Segunda Forma Normal, além de ser necessário estar dentro da Primeira Forma, também tem a regra que os atributos não chaves, devem depender exclusivamente da chave primária da tabela, as colunas não dependentes devem ser retiradas da tabela e colocadas em uma nova tabela utilizando esses dados.

Exemplo:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Id\_prof | Id\_curso | Salário | Desc\_curso |
| Carlos | 01 | 001 | R$ 5.500 | Curso de desenhos, aprenda técnicas para desenhar animes e muito mais |
| roberto | 02 | 002 | R$ 3.500 | Curso de Engenharia Civil, seja um engenheiro renomado neste rumo |

Neste exemplo, o atributo “Desc\_curso” não depende unicamente da chave primária que é o “id\_prof”, ela também depende do “id\_curso”, desta forma, devemos criar uma nova tabela:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Id\_prof | Id\_curso | Salário |
| Carlos | 01 | 001 | R$ 5.500 |
| Roberto | 02 | 002 | R$ 3.500 |

|  |  |
| --- | --- |
| Id\_curso | Desc\_curso |
| 001 | Curso de desenhos, aprenda técnicas para desenhar animes e muito mais |
| 002 | Curso de Engenharia Civil, seja um engenheiro renomado neste rumo |

Terceira Forma

Para aplicarmos a Terceira Forma normal, além de termos aplicado a segunda, devemos eliminar campos repetidos de outras tabelas que não estejam ligados ao seu “tema”.

“Todas as colunas devem ter dependência funcional com a totalidade de cada chave candidata e nada mais além do que essas chaves candidatas.”

Exemplo:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N\_pedido | Codigo\_prod | Quant | Valor\_unit | Subtotal |
| 1005 | 1-932 | 5 | 1.500,00 | 7.500,00 |
| 1006 | 1-956 | 3 | 350,00 | 1.050,00 |
| 1007 | 1-922 | 1 | 190,00 | 190,00 |
| 1008 | 1-908 | 6 | 980,00 | 5.800,00 |

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\_prod | Quant | Valor\_unit |
| 1005 | 1-932 | 5 | 1.500,00 |
| 1006 | 1-956 | 3 | 350,00 |
| 1007 | 1-922 | 1 | 190,00 |
| 1008 | 1-908 | 6 | 980,00 |

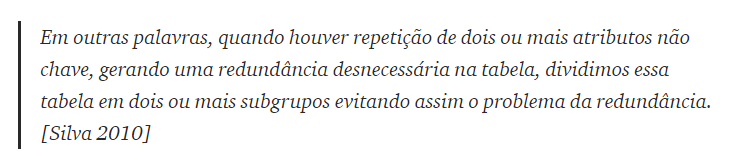
FNBC (Forma Normal Boyce-Codd)

Esta Forma exige que não seja possível acessar uma chave candidata por meio de uma outra chave candidata, e que não exista dependência funcional não trivial de atributos além de um superconjunto de uma chave candidata.

Nesta forma não é permitido chegar-se a uma chave candidata com base em alguma outra chave candidata por meio de dependências funcionais.

Quarta Forma

A Quarta Forma tem o objetivo de eliminar as redundâncias dos valores da tabela:



Ou seja, caso um valor determinado consiga ser obtido por operações com outros valores, não é necessário existir um campo para receber esse valor:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Código | Produto | Quantidade | Preço | Total |
| 0548 | Batata | 5 | R$ 2,90 | R$ 14,5 |
| 0896 | Maçã | 10 | R$ 1,15 | R$ 11,5 |

No exemplo acima, podemos observar que o total, podemos obter pela multiplicação do preço pela quantidade, desta forma, o campo total pode ser eliminado da tabela, pois podemos obter o resultado pela multiplicação:

|  |  |  |  |
| --- | --- | --- | --- |
| **Código** | **Produto** | **Quantidade** | **Preço** |
| 0548 | Batata | 5 | R$ 2,90 |
| 0896 | Maçã | 10 | R$ 1,15 |

Quinta Forma

A Quinta Forma Normal não se difere muito da Quarta Forma Normal, se não houver uma constante simétrica que atue como uma regra de mundo entre as tabelas, ela não se diferencia da quarta forma, se não houver essa constante, ela é idêntica a 4° forma.

|  |  |  |
| --- | --- | --- |
| vendedor | empresa | produto |
| 1 | Samsung | SSD |
| 1 | Samsung | Fones de Ouvido |
| 1 | Apple | Fones de Ouvido |

Desta Forma, ele está na 4° Forma, porém, se uma regra de mundo for aplicada, se um vendedor vende um tipo de produto e também vende para determinada empresa, então este vendedor vende aquele produto para aquela empresa. Ou seja, se ele vende discos rígidos e vende para a Apple, automaticamente vende discos rígidos da Apple.

Se esta regra de mundo for verdadeira, então não estará mais na 5° Forma, para que se enquadre novamente na quinta forma, então teremos que separar o que ele vende de pra quem ele vende.

|  |  |  |
| --- | --- | --- |
| vendedor | empresa | produto |
| 1 | Samsung | SSD |
| 1 | Samsung | Fones de Ouvido |
| 1 | Apple | SSD |
| 1 | Apple | Fones de Ouvido |

Porém, esse esquema gera repetição de dados, para resolvermos isso, podemos separar em um produto cartesiano.

|  |  |
| --- | --- |
| vendedor | empresa |
| 1 | Samsung |
| 1 | Apple |

|  |  |
| --- | --- |
| vendedor | produto |
| 1 | SSD |
| 1 | Fones de Ouvido |

|  |  |
| --- | --- |
| empresa | produto |
| Samsung | SSD |
| Samsung | Fones de Ouvido |
| Apple | SSD |
| Apple | Fones de Ouvido |

Deste modo, elimina-se a repetição de dados separando em três tabelas, empresas para qual ele vende, produtos que o vendedor vende e os produtos que a companhia vende.