**Grupo 2**

**Tema:** Normalização

**Professores:** Saulo e Lucas

**Repositório GIT:**

<https://github.com/Marcaum04/Normalizacao-G2M>

**Trello:**

<https://trello.com/b/daFSw7Bg/normaliza%C3%A7%C3%A3o>

**Participantes:**

* Gustavo Borges
* Levi Bueno
* Lucas Medina
* Lívia Negrini
* Marcos Vinicius
* Matheus Araújo

**Emails:**

[sgustavo.borges10@gmail.com](mailto:sgustavo.borges10@gmail.com)

[lbds.bueno@gmail.com](mailto:lbds.bueno@gmail.com)

[lucas.m3dina@gmail.com](mailto:lucas.m3dina@gmail.com)

[liviasnegrini2004@gmail.com](mailto:liviasnegrini2004@gmail.com)

[mvinicius.oliveira04@gmail.com](mailto:mvinicius.oliveira04@gmail.com)

[matheusaraujoms.machado@gmail.com](mailto:matheusaraujoms.machado@gmail.com)

**Apresentação:**

* Conteúdo desenvolvido
* Exemplo prático
* Exercício
* Material de apoio

**Entrega:**

* Link do repositório do GitHub com todos os

conteúdos desenvolvidos

* Link do quadro do Trello com todo o planejamento

das atividades

**Roteiro:**

Cada um com duas páginas, todos pesquisam tudo - Escrever o que é, por que usar?(benefícios), quais são as formas normais(explicar).

**Gustavo Borges Folha 1**

**O que é:**

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 de atualização, inserção e exclusão de registros. Criando assim relações mais bem estruturadas

A normalização de tabelas tem por objetivo principal resolver problemas de atualização de bases de dados, minimizando redundâncias. Ele pode ser visto como o processo no qual são eliminados esquemas de relações (tabelas) não satisfatórios, decompondo-os, através da separação de seus atributos em esquemas de relações menos complexas, mas que satisfaçam as propriedades desejadas.

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.

*O processo de normalização compreende o uso de um conjunto de regras, chamados de formas normais. Ao analisarmos o banco de dados e verificarmos que ele respeita as regras da primeira forma normal, então podemos dizer que o banco está na “primeira forma normal”. Caso o banco respeite as primeiras três regras, então ele está na “terceira forma normal”. Mesmo existindo mais conjuntos de regras para outros níveis de normalização, a terceira forma normal é considerada o nível mínimo necessário para grande parte das aplicações. [Microsoft 2007]*

O processo de normalização como foi proposto inicialmente por Codd conduz um esquema de relação através de uma bateria de testes para certificar se o mesmo está na 1ª, 2ª e 3ª Formas Normais. Cada etapa ou teste corresponde uma determinada forma normal, que representa um progressivo refinamento na estrutura das tabelas. Assim, uma tabela que se encontra na Terceira Forma Normal é considerada mais normalizada (mais "enxuta", pode-se dizer) que uma tabela que se encontra apenas na Segunda Forma Normal.

A normalização possui caráter organizativo e pode ocorrer durante a concepção do modelo conceitual, durante a derivação do modelo lógico para o relacional, ou após a derivação do modelo lógico.

**As principais características de uma base de dados normalizada são:**

* **Geração de aplicações mais estáveis.**
* **Aumento do número de tabelas utilizadas.**
* **Diminuição dos tamanhos médios das tabelas.**

Ao efetuar o processo de normalização, os dados cadastrados no banco de dados ficarão organizados de uma forma melhor e na maioria das vezes também ocuparão menos espaço físico. Entretanto, o processo de normalização também sempre faz aumentar o número de tabelas e em muitos casos pode ser uma tarefa difícil de ser realizada. Além disso, bancos de dados normalizados além do necessário podem ter desempenho ruim e/ou complexidade excessiva.

Normalização é uma ferramenta usada no projeto lógico que serve para reestruturar tabelas e atributos, reduzindo assim redundâncias e permitindo o correto crescimento do banco de dados. Por meio dela que bancos com muita movimentação garantem sua integridade após remoção, inserção e alteração dos dados.

**O processo de normalização conta com 6 formas:**

* **1° Forma Normal**
* **2° Forma Normal**
* **3° Forma Normal**
* **FNBC (Forma normal de Boyce e Codd)**
* **4° Forma Normal**
* **5° Forma Normal**

A partir da 3° forma normal diz-se que o banco de dados já se encontra normalizado. A FNBC, a 4FN e a 5FN são usadas para refinar ainda mais o banco. No entanto alguns projetos decidem parar na 3FN pois as outras formas, dependendo da situação, podem exigir um pouco mais de processamento.

**Por que usar? Benefícios**

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.

**Benefícios da normalização:**

* **Estabilidade do modelo lógico: capacidade de um modelo manter-se inalterado face às mudanças que venham a ser percebida 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.**

**Aplicando as regras descritas a seguir, é possível garantir um banco de dados mais íntegro, sem redundâncias e inconsistências.**

**Levi Bueno Folha 1**

**Levi Bueno Folha 2**

**Lucas Medina**

**Definição**

A normalização de tabelas tem por objetivo principal resolver problemas de atualização de bases de dados, minimizando redundâncias. Ele pode ser visto como o processo no qual são eliminados esquemas de relações (tabelas) não satisfatórios, decompondo-os, através da separação de seus atributos em esquemas de relações menos complexas, mas que satisfaçam as propriedades desejadas.

**Benefícios da normalização:**

* 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.

# **Primeira forma normal (1NF)**

A primeira regra para eliminar as anomalias é:

**“Não devem existir colunas multivaloradas.”**

Uma coluna multivalorada é uma coluna na qual é possível armazenar-se mais de um valor por registro.

**Segunda forma normal (2FN)**

A segunda forma normal é aquela que diz que:  
**“Todas as colunas devem ter dependência funcional com a totalidade de cada chave candidata.”**

Na maioria dos casos por "*cada chave candidata*", entenda-se por "*com a chave primária*", exceto se houver mais do que uma chave candidata.

Além disso, para que uma tabela esteja na segunda forma normal, ela deve estar antes de tudo, na primeira forma normal.

Uma coluna está em dependência funcional com a chave primária quando ela é determinada no domínio da aplicação por meio da chave primária. Uma coluna não tem dependência funcional com a chave primária quando ela é definida de forma independente da chave primária ou quando ela é definida a partir de alguma outra coluna que não seja a chave primária.

Uma dependência funcional pode ser dita estar na totalidade da chave primária quando todos os campos da chave primária são necessários para estabelecer-se a relação de dependência. No caso de a chave primária ser composta, é possível ter uma dependência parcial.

# **Terceira forma normal (3FN)**

A terceira forma normal é aquela que diz que:

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

Novamente, na maioria dos casos por "*cada chave candidata*", entenda-se por "*com a chave primária*", exceto se houver mais do que uma chave candidata. Se a única chave candidata existente for a chave primária, isso ficaria assim:

**“Todas as colunas devem ter dependência funcional com a totalidade da chave primária e nada mais além da chave primária.”**

Além disso, para uma tabela estar na terceira forma normal, ela deve estar primeiramente na segunda forma normal (e também na primeira). A parte de depender da totalidade de cada chave candidata é abordada na segunda forma normal, então o foco aqui é depender de nada mais que essas chaves.

# **Quarta forma normal (4FN)**

É raro encontrar-se casos de tabelas que estejam na terceira forma normal, mas não na quarta. A quarta forma concerne em anomalias existentes na relação entre diferentes colunas da chave primária, e só se aplica em tabelas com chaves primárias compostas por três colunas ou mais.

Tabelas que representem relacionamentos ternários, quaternários ou n-ários de muitos-para-muitos são locais onde vale dar-se uma olhada em possíveis violações da 4FN.

Para uma tabela estar na quarta forma normal, ela tem que primeiramente estar também na terceira forma normal. A forma normal de Boyce-Codd não é necessária.

# **Quinta forma normal (5FN)**

A quinta forma normal é mais restrita que a quarta, e também se aplica às tabelas com 3 ou mais colunas na chave primária. Novamente, para que a quinta forma normal seja atingida, é necessário atingir-se a quarta forma normal primeiramente.

Quando um campo (atributo) está em outra tabela sem a necessidade de estar na tabela pesquisada. Não há a perda de nenhuma informação.

# **Chaves candidatas e superchaves**

Antes de prosseguir com as demais formas normais, faz-se necessário introduzir-se os conceitos de chaves candidatas e superchaves.

A chave primária (*primary key*, ou apenas **PK**) é aquele conjunto de colunas que serve para identificar a tupla de uma forma única (pode ser só uma coluna, ou podem ser duas ou mais). É importante que o projetista do banco de dados saiba identificar quais são as colunas mais apropriadas para serem eleitas como parte da chave primária. Muitas vezes, uma coluna com um código numérico sequencial pode servir para ser a chave primária, mas há várias situações onde esse não é o caso.

Às vezes, há mais do que um conjunto de colunas que poderia ser chave primária. Cada um desses conjuntos é chamado de chave candidata. Por exemplo, em uma tabela Pessoa que tenha os campos CPF, RG, Estado e Nome, tanto o CPF quanto o RG junto com o Estado são chaves candidatas. Assim, é possível chegar-se ao Nome a partir do CPF, mas também é possível chegar-se ao Nome a partir do RG e do Estado.

Qualquer conjunto de colunas que tenha como subconjunto, uma chave candidata é denominado de superchave.

**Lívia Negrini Folha 1**

**O que é?**

Normalização é o processo de organização de dados em um banco de dados. Isso inclui a criação de tabelas e o estabelecimento de relações entre essas tabelas de acordo com as regras projetadas para proteger os dados e tornar o banco de dados mais flexível, eliminando a redundância e a dependência inconsistente.

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.

**Por que usar?**

Normalização de banco de dados é um conjunto de regras que visa, principalmente, a organização de um projeto de banco de dados para reduzir a redundância de dados, aumentar a integridade de dados e o desempenho.

**Primeiro formulário normal**

* Eliminar grupos repetidos em tabelas individuais.
* Crie uma tabela separada para cada conjunto de dados relacionados.
* Identifique cada conjunto de dados relacionados com uma chave primária.

Não use vários campos em uma única tabela para armazenar dados semelhantes. Por exemplo, para rastrear um item de inventário que pode vir de duas fontes possíveis, um registro de inventário pode conter campos para o Código do Fornecedor 1 e o Código do Fornecedor 2.

**Segundo formulário normal**

* Crie tabelas separadas para conjuntos de valores que se aplicam a vários registros.
* Relacione essas tabelas com uma chave estrangeira.

Os registros não devem depender de nada além da chave primária de uma tabela (uma chave composta, se necessário). Por exemplo, considere o endereço de um cliente em um sistema de contabilidade. O endereço é necessário pela tabela Clientes, mas também pelas tabelas Pedidos, Envio, Faturas, Contas A receber e Coleções. Em vez de armazenar o endereço do cliente como uma entrada separada

em cada uma dessas tabelas, armazene-o em um só lugar, na tabela Clientes ou em uma tabela Endereços separada.

**Lívia Negrini Folha 2**

**Terceiro formulário normal**

* Elimine campos que não dependem da chave.

Os valores em um registro que não fazem parte da chave desse registro não pertencem à tabela. Em geral, sempre que o conteúdo de um grupo de campos pode se aplicar a mais de um único registro na tabela, considere colocar esses campos em uma tabela separada.

**Marcos Vinícius**

**O que é?**

**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 repetição no banco.**

**Por que usar?**

**Tal procedimento é feito a partir da identificação de uma anomalia em uma relação, decompondo-as em relações melhor estruturadas. Um banco de dados dentro dos padrões de normalização reduz o trabalho de manutenção e ajuda a evitar o desperdício do espaço de armazenamento.**

**Primeira Forma Normal**

**Uma relação está na primeira forma normal quando todos os atributos dos valores em um registro que não fazem parte da chave deste registro não pertencem à tabela. Em geral, sempre que o conteúdo de um grupo de campos pode se aplicar a mais de um único registro na tabela, considere colocar esses campos em uma tabela separadas contém apenas um valor correspondente, singular e não existem grupos de atributos repetidos**

**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.**

**Terceiro formulário normal**

**Os valores em um registro que não fazem parte da chave deste registro não pertencem à tabela. Em geral, sempre que o conteúdo de um grupo de campos pode se aplicar a mais de um único registro na tabela, considere colocar esses campos em uma tabela separada.**

Problemas de Inserção - **Só é possível inserir um cliente se o mesmo adquirir um produto - Só é possível inserir um produto se algum cliente o adquirir**

**Problemas de alteração**

**- Para atualizar o telefone do cliente todos os outros registros deverão ser atualizados. - Para atualizar o preço do produto todos os registros desse mesmo produto deverão ser atualizados.**

**Problemas de exclusão**

**- Se os produtos adquiridos por algum cliente forem excluídos, os dados cadastrais do mesmo se perderão.**

**Matheus Araújo Folha 1**

**Definição**

Normalização de banco de dados é um conjunto de regras que visa, principalmente, a organização de um projeto de banco de dados para reduzir a redundância de dados, aumentar a integridade de dados e o desempenho. Para normalizar o banco de dados, deve-se examinar as colunas (atributos) de uma entidade e as relações entre entidades (tabelas), com o objetivo de se evitar anomalias observadas na inclusão, exclusão e alteração de registros

**Objetivo**

Um objetivo básico da primeira forma normal, definida por Codd em 1970, era permitir dados serem questionados e manipulados usando uma "sub-linguagem de dados universal" atrelada à lógica de primeira ordem. Questionando e manipulando dados em uma estrutura de dados não normalizada.

**Por que utilizar?**

Normalização é o processo de organização de dados em um banco de dados. Isso inclui a criação de tabelas e o estabelecimento de relações entre essas tabelas de acordo com as regras projetadas para proteger os dados e tornar o banco de dados mais flexível, eliminando a redundância e a dependência inconsistente

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.

**Formas normais**

As formas normais são sequenciais, ou seja, se um banco se encontra na terceira forma normal, isso também significa que o mesmo está na segunda e também na primeira. Por isso devemos sempre começar a normalização pela primeira forma normal, para que não hajam problemas mais a frente na nossa normalização

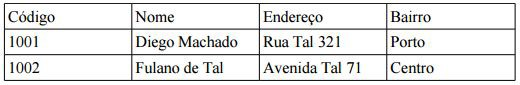
**Matheus Araújo Folha 2**

**Primeira Forma Normal**

Uma relação está na primeira forma normal quando todos os atributos contém 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.

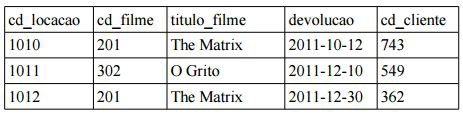


Analisando o exemplo acima, podemos observar dois problemas: temos uma pessoa com dois números de telefone e um endereço com diferentes valores, a rua e o bairro. A fim de normalizar, teremos que colocar cada informação em uma coluna diferente e criar uma nova tabela relacionando a pessoa a seus números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



**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



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.

**Matheus Araújo Folha 3**

**Terceira Forma Normal**

**Referências Bibliográficas:**

[**MORE | Início**](https://more.ufsc.br/)

[**Descrição de normalização do banco de dados - Office**](https://docs.microsoft.com/pt-br/office/troubleshoot/access/database-normalization-description)

[**Recapitulando**](http://www.dsc.ufcg.edu.br/~pet/jornal/maio2011/materias/recapitulando.html)

[**Normalização em Bancos de Dados. Este artigo descreve o que é o processo… | by Diego Machado**](https://medium.com/@diegobmachado/normaliza%C3%A7%C3%A3o-em-banco-de-dados-5647cdf84a12)

[**Descrição de normalização do banco de dados - Office**](https://docs.microsoft.com/pt-br/office/troubleshoot/access/database-normalization-description)

[**5 - Normalização - Banco de Dados I**](https://sites.google.com/site/fkbancodedados1/normalizacao)

[**Aprenda na Prática uma Normalização Banco de Dados**](http://aprendaplsql.com/modelagem-de-dados/normalizacao-banco-de-dados/)

[**O que é normalização de banco de dados?**](https://pt.stackoverflow.com/questions/151323/o-que-%C3%A9-normaliza%C3%A7%C3%A3o-de-banco-de-dados)

[**Normalização de dados e as formas normais**](https://www.luis.blog.br/normalizacao-de-dados-e-as-formas-normais.html)

[**formas normais**](http://www.facom.ufu.br/~wendelmelo/bd201801/formas_normais.pdf)

[**Banco de Dados**](https://www.ime.usp.br/~andrers/aulas/bd2005-1/aula11.html)

[**Aprendendo como se faz a normalização em um Banco de Dados**](http://spaceprogrammer.com/bd/normalizando-um-banco-de-dados-por-meio-das-3-principais-formas/)