



# **BANCO DE DADOS I**

Rodrigo Kiyoshi Saito / [rodrigok@anchieta.br](mailto:rodrigok@anchieta.br)

# Tópicos abordados

- Formas Normais

# FORMAS NORMAIS

- Segundo BATTIST (2005, P. 16), o objetivo da normalização é evitar os problemas provocados por falhas no projeto do Banco de Dados, bem como eliminar a “mistura” de assuntos e as correspondentes redundâncias de dados.
- A normalização de tabelas é utilizada para tentar detectar erros no projeto das tabelas e atributos de cada tabela e corrigir estes erros, antes da criação e utilização do Banco de Dados.

# FORMAS NORMAIS

- É bem mais fácil (e barato), corrigir os erros na fase de projeto do que depois que o Banco de Dados já estiver em uso.
- Uma “Regra de ouro” que devemos observar quanto ao projeto de banco de dados é a de “não misturar assuntos em uma mesma tabela”.

# FORMAS NORMAIS

- GUIMARAES (2003, p.81) complementa que existe um considerável aparato teórico por trás dos conceitos de normalização de relações. As razões para estudá-los, no entanto, são de ordem prática. Eles nos vão ajudar a projetar base de dados com menos possibilidade de inconsistências e menos redundâncias de informação.

# FORMAS NORMAIS

- Eles também vão ajudar a determinar com mais precisão certos tipos de restrições sobre atributos de uma relação assim como a determinação das suas possíveis chaves.

# FORMAS NORMAIS

- Por exemplo, na tabela Clientes, devemos colocar somente campos relacionados com o assunto Clientes. Não devemos misturar campos relacionados com outros assuntos, tais como pedidos, produtos etc. Essa “mistura de assuntos” em uma mesma tabela acaba por gerar repetição desnecessária dos dados bem como inconsistências dos dados.

# FORMAS NORMAIS

- O processo de normalização aplica uma série de regras sobre as tabelas de banco de dados, para verificar se estas estão corretamente projetadas. Embora existam cinco formas normais (ou regras de normalização), na prática usamos um conjunto de três formas normais.



# FORMAS NORMAIS

- Frequentemente, após a aplicação das regras de normalização, algumas tabelas acabam sendo divididas em duas ou mais, o que no final gera um número maior de tabelas do que o número de tabelas originalmente projetado.

# FORMAS NORMAIS

- Este processo causa a simplificação dos atributos de uma tabela, colaborando significativamente para a estabilidade do projeto de banco de dados, reduzindo-se as necessidades de manutenção e alterações, após o banco ter sido colocado em produção.

# ***Primeira Forma Normal***

- BATTIST diz que a Regra: “Uma tabela está na Primeira Forma Normal quando seus atributos não contem grupos de repetição”. Por isso dizemos que uma tabela que possui grupos de repetição não está na Primeira Forma Normal.

# Primeira Forma Normal

- Exemplo de uma tabela que não está normalizada na 1 FN (BATTIST, 2005, p.17)

AlunosNaoNormalizado

NomeDoAluno
NumeroDaMatricula
DataNascimento
Serie
NomeDoPai
NomeDaMae

# ***Primeira Forma Normal***

- GUIMARAES (p.84) diz que a 1FN é considerada como a própria definição do MR: Toda relação está em 1FN, isto é, não possui atributos multivalorados nem relações aninhadas e será tomada como implícita.

# ***Primeira Forma Normal***

- Podemos notar que uma tabela com esta estrutura apresenta diversos problemas. Por exemplo, se um casal tiver mais do que um filho, teríamos que digitar o nomes do pai e da mãe diversas vezes, tantas quantos forem os filhos. Isso forma um grupo de repetição.

# ***Primeira Forma Normal***

- Pode ser que, por erro de digitação, o nome dos pais não apareça exatamente igual todas as vezes, o que pode acarretar problemas na hora de fazer pesquisas ou emitir relatórios. Este problema ocorre porque misturamos assuntos em uma mesma tabela. Colocamos as informações dos pais e dos filhos em uma mesma tabela. (BATTIST, p. 17)

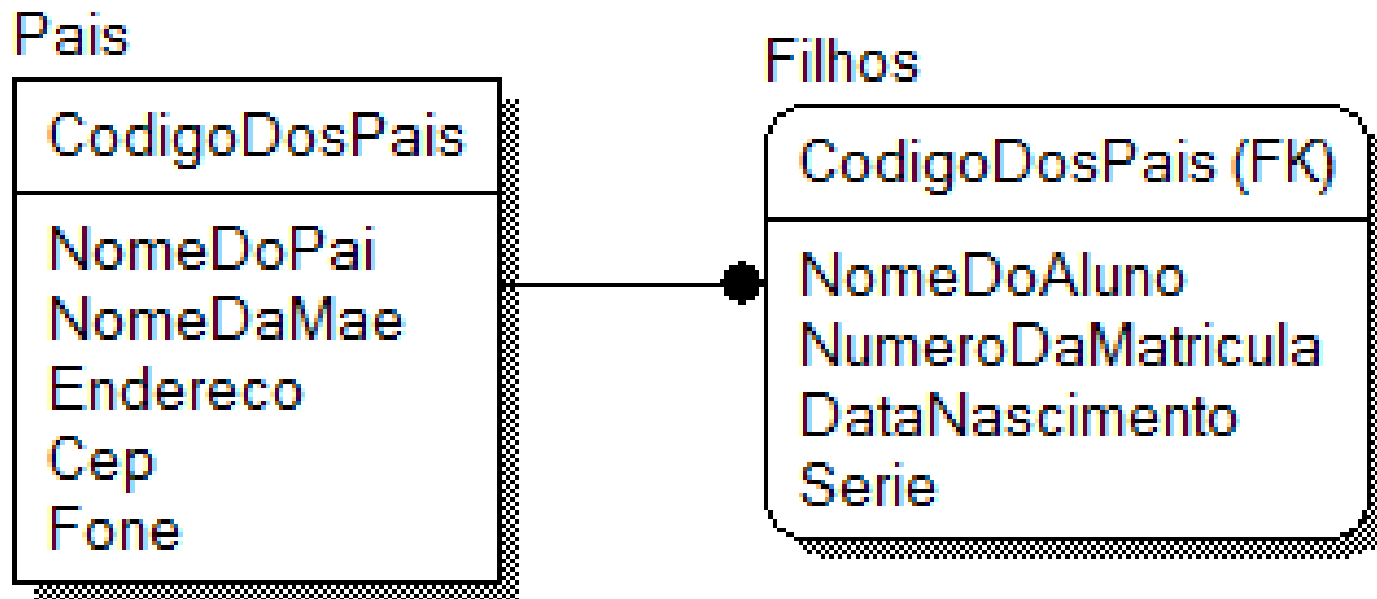
# ***Primeira Forma Normal***

- A solução para este problema é simples: criamos uma tabela separada para a informação dos pais e relacionamos a tabela Pais com a tabela Filhos através de um relacionamento do tipo Um para Muitos, ou seja, um casal pode ter vários filhos.



# Primeira Forma Normal

- As Tabelas Pais e Filhos estão na Primeira Forma Normal (BATTIST, 2005, p.17)



# ***Primeira Forma Normal***

- As duas tabelas resultantes da aplicação da Primeira Forma Normal, Pais e Filhos, estão na Primeira Forma Normal. A tabela original, a qual misturava informações de pais e filhos, não está na Primeira Forma Normal.

# ***Primeira Forma Normal***

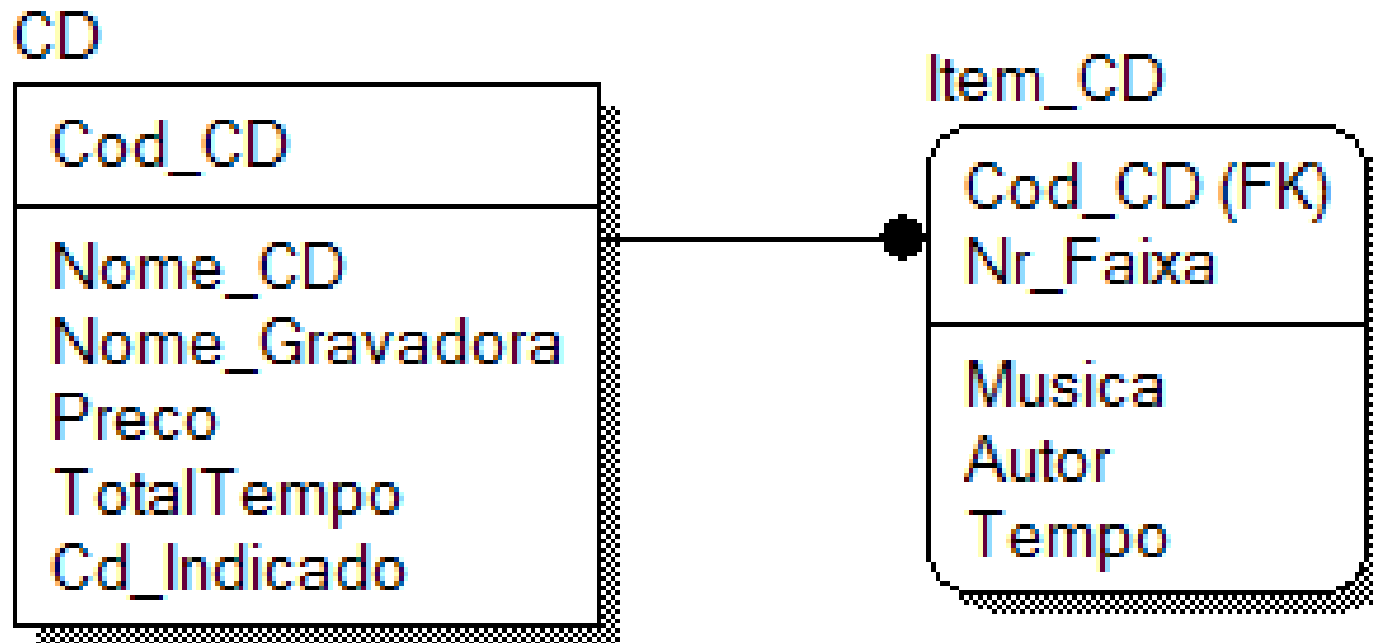
- OLIVEIRA (2002, p.53) complementa que a solução para este caso é que devemos separar a informação que se repete em uma nova entidade. Devemos ainda levar a chave primaria da entidade original para a nova entidade gerada (caso contrário, não haverá como relacionar as informações das duas entidades). Feito isso, criaremos a nova chave para a nova entidade.

# ***Primeira Forma Normal***

- Normalmente podemos localizar um campo que, unido à chave da entidade original, formará a chave da nota tabela (nesse caso, chave concatenada), ou podemos criar um campo para esse fim, caso não exista, Há a possibilidade de criarmos simplesmente uma nova chave não concatenada e independente da entidade original. É uma questão de preferência.

# Primeira Forma Normal

- Exemplo:



# ***Primeira Forma Normal***

- Utiliza-se o campo Numero da Faixa como atributo-chave junto com o CodigoDoCD, pois não poderá haver duas faixas com o mesmo número em um único CD. Essa é, portanto, uma chave concatenada.

# ***Segunda Forma Normal***

- Podemos aplicar a Segunda Forma Normal quando tivermos uma chave primária composta. Neste caso, devemos observar se todos os campos, que não fazem parte da chave primária composta, dependem de todos os campos que compõem a chave primária composta. Se algum campo depender somente de parte da chave primária composta, então este campo deve pertencer a outra tabela. (BATTIST, 2005, P.17)

# Segunda Forma Normal

- Tabela que não está na segunda forma normal (BATTIST, 2005, p.18)

naoEstaNa2FN

NumeroDaMatricula CodigoDoCurso
Avaliacao DescricaoDoCurso



# ***Segunda Forma Normal***

- A chave primária composta é formada pela combinação dos campos NúmeroDaMatricula e CódigoDoCurso.
- O campo Avaliação depende tanto do CódigoDoCurso quanto Do NúmeroDaMatricula (cada aluno – representado por sua matricula, tem uma nota em cada disciplina – representada pelo campo CódigoDoCurso), porém o campo DescriçãoDoCurso depende apenas do CódigoDoCurso (a descrição do curso não tem relação com o NúmeroDaMatricula).

# ***Segunda Forma Normal***

- Com isso, temos um campo que não faz parte da chave primária composta e depende apenas de um dos campos que compõem a chave primária composta. Assim podemos dizer que esta tabela não está na Segunda Forma Normal. (BATTIST, 2005, P.18)

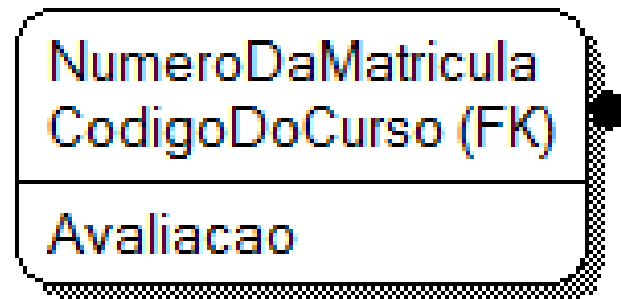
# ***Segunda Forma Normal***

- A resolução para este problema também é simples: dividimos a tabela, que não está na Segunda Forma Normal, em duas outras tabelas, conforme indicado pela figura abaixo, sendo que as duas tabelas resultantes estarão na segunda forma normal.

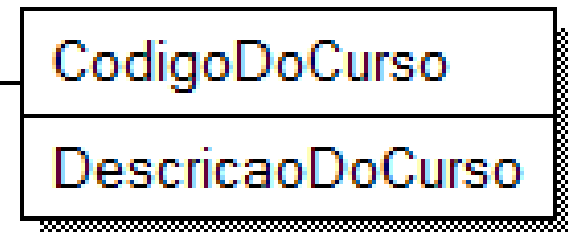
# Segunda Forma Normal

- Duas tabelas que estão na Segunda Forma Normal:

Avaliacaoes



Cursos



# ***Segunda Forma Normal***

- OLIVEIRA (2002, p.56) diz que uma entidade está na segunda forma normal quando todos os seus atributos não chave dependem unicamente da chave. Assim, deve-se perguntar a cada atributo, que não seja a chave da entidade, se ele depende apenas da chave da entidade. O que se procura com isso é medir o grau de dependência entre os atributos.

# ***Segunda Forma Normal***

- Apenas coisas semelhantes podem ser unidas em grupos semelhantes (entidades nada mais são do que um grupo ou conjunto). A solução sugerida para esse problema é ao identificarmos uma situação como a descrita anteriormente, devemos separar os atributos independentes e criar ou identificar dentre os atributos separados uma nova chave para esta nova entidade.

# ***Segunda Forma Normal***

- Essa chave deve ser mantida na entidade original como o atributo de relacionamento entre ambas as entidades. Dessa forma, não se perde qualquer informação no modelo.

## ***Segunda Forma Normal***

- No exemplo anterior, a gravadora, o Autor e a Musica são independentes de suas entidades, CD e Item\_CD. Veja que já uma enorme vantagem ao separarmos esses atributos independentes, visto que se não o fizéssemos, cada alteração em uma das informações deveria ser estendida a todas as linhas da entidade.



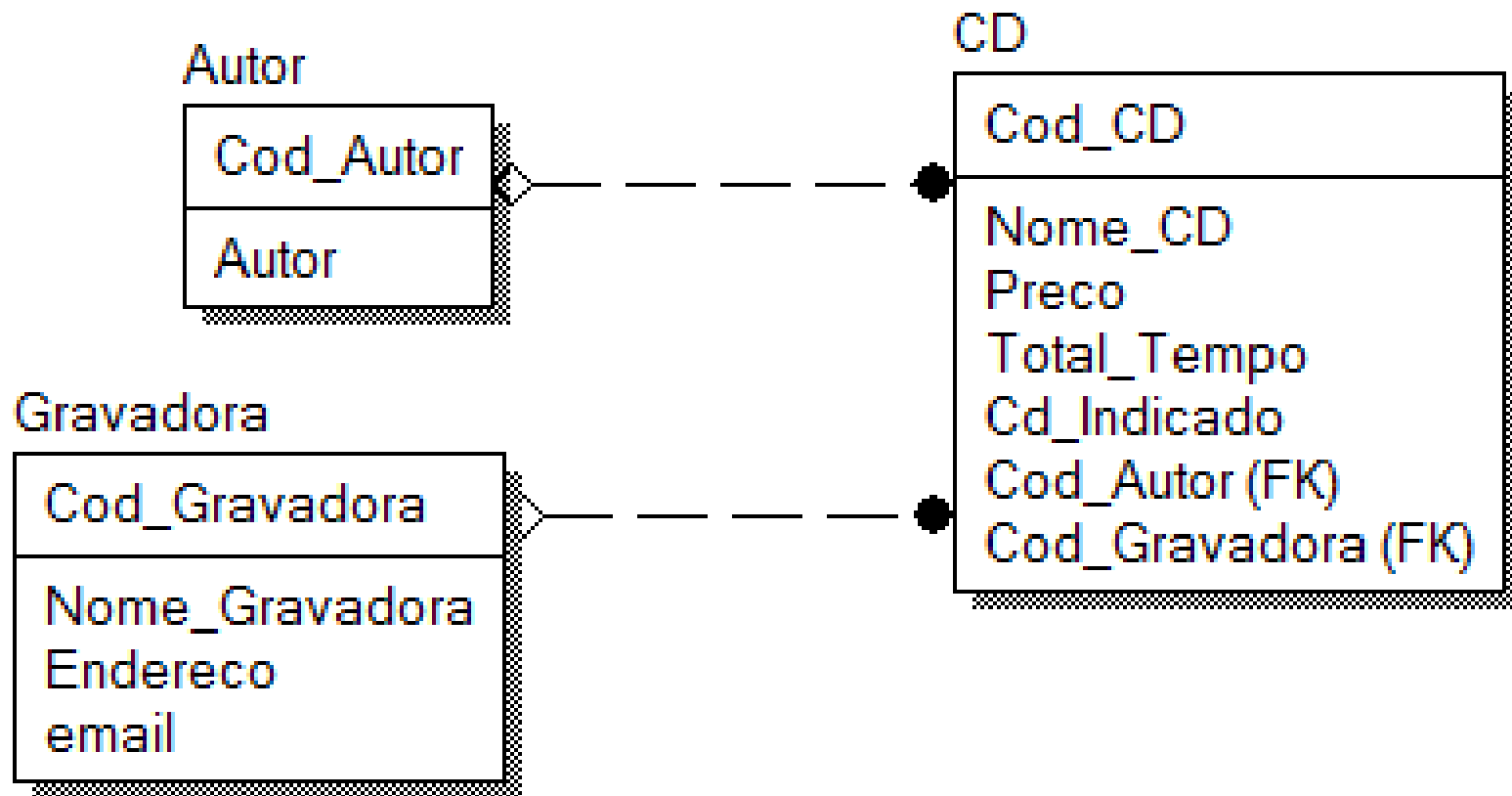
## ***Segunda Forma Normal***

- É muito mais fácil fazer isso apenas na nova entidade criada. Como a relação se dá por meio do campo-chave, não há necessidade de alterar todas as ocorrências em CD ou Item\_CD, apenas nas entidades Autor, Gravadora ou Musica.

# ***Segunda Forma Normal***

- A inclusão de novos atributos também é facilitada. Atributos que venham a ser relevantes serão acrescentados diretamente na entidade Autor (como Data de Nascimento) ou Gravadora (como Endereço e Home Page)

# Segunda Forma Normal



# ***Terceira Forma Normal***

- Segundo BATTIST (p.21), na definição dos campos de uma tabela podem ocorrer casos em que um campo não seja dependente diretamente da chave primária, ou de parte dela, mas sim dependente de um outro atributo constante na tabela, atributo este que não seja a chave primária.

# ***Terceira Forma Normal***

- Quando isso ocorre, dizemos que a tabela não está na Terceira Forma Normal, conforme indicado pela figura seguinte:

# Terceira Forma Normal

- Uma tabela que não está na Terceira Forma Normal:

## FuncionariosNaoNormalizado

NumeroDaMatricula
NomeFuncionario CodigoDoCargo DescricaoDoCargo

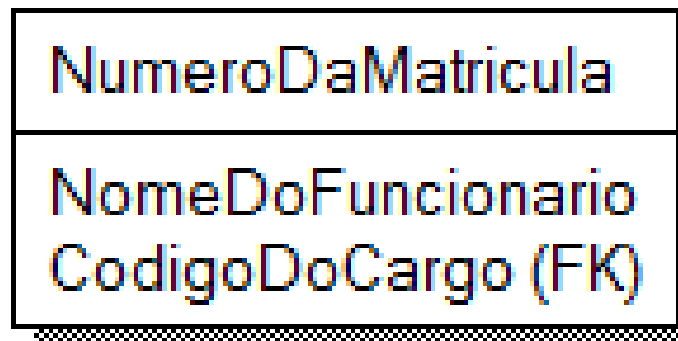
# ***Terceira Forma Normal***

- Observe que o campo DescriçãoDoCurso depende apenas do campoCodigoDoCurso, o qual não faz parte da chave primária. Por isso dizemos que esta tabela não está na Terceira Forma Normal.
- A solução para este caso também é simples. Novamente basta dividir a tabela em duas outras, conforme indicado pela figura abaixo. As duas tabelas resultantes estão na Terceira Forma Normal.

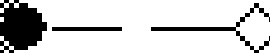
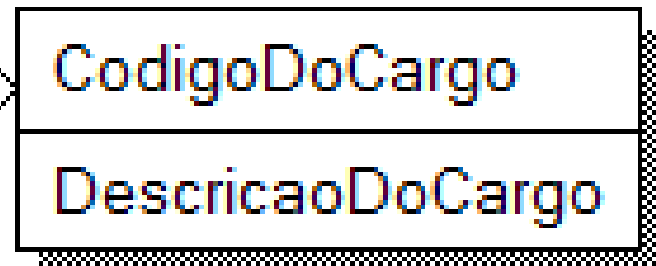
# Terceira Forma Normal

- Duas tabelas que estão na Terceira Forma Normal:

Funcionarios



Cargos





# ***Terceira Forma Normal***

- OLIVEIRA (p.60) diz que uma entidade está terceira forma normal quando todos os seus atributos não chave não dependem de nenhum outro atributo não chave. Utilizando palavras mais simples, pode-se dizer que um atributo não deve depender de outro atributo. Isso ocorre normalmente em cálculos matemáticos ou em atributos “perdidos” na entidade errada.

# ***Terceira Forma Normal***

- Se guardamos eles são resultados de uma operação entre dois atributos. Se guardarmos esse valor, estaremos apenas ocupando espaço e abrindo a possibilidade de termos uma informação inconsistente no banco de dados;

## ***Terceira Forma Normal***

- Exemplo: No nosso exemplo, o atributo TempoTotal é dependente de outro atributo não chave: Tempo, na entidade musica. Ao somarmos o tempo de cada faixa do CD, teremos o tempo total de gravação do CD. Esse campo deve ser eliminado, sem perda de informação para o modelo de dados.

# Terceira Forma Normal

CD

Cod\_CD

Nome\_CD

Cod-Gravadora

Preco

CD\_Indicado

# ***Exercício - Formas Normais***



- Dado a seguinte figura, defina as tabelas utilizando as seguintes formas normais: 1FN, 2FN e 3FN:

# Exercício - Formas Normais



NR PEDIDO DE COMPRAS: 151515

DATA: 15/05/2005

ORIGEM: DEPTO ADMINISTRATIVO

FUNC SOLICITANTE: MARIA CRISTINA

DESTINO: DEPTO COMPRAS

FUNC RESPONSÁVEL: JOAO FERREIRA

ITEM	MATERIAL	QUANTIDADE
01	LAPIS	10
02	CANETA	05
03	BORRACHA	02
04	APONTADOR	01
05	MIDIA DVD VIRGEM	10

TOTAL QUANTIDADE: 28

Ficha exemplo – Exercício 01