



Universidade
FIRJAN |  INFORMA, FORMA, TRANSFORMA.




Banco de Dados

Fabício Curvello Gomes

Universidade
FIRJAN |  INFORMA, FORMA, TRANSFORMA.



Normalização




O que é Normalização?


É um conjunto de regras destinadas ao refinamento de uma modelagem, de forma que esta atenda às normas estabelecidas em modelagem de bancos de dados relacionais.

A normalização tem como principais objetivos:

- Eliminar as probabilidades de redundâncias de informações, ou seja, quando uma mesma informação pode ser obtida em mais de um ponto do banco de dados (repetição de informações)
- Eliminar as probabilidades de inconsistências de informações que acontecem quando a busca por uma determinada informação resulta em respostas distintas, não certificando exatamente qual é a verdadeira informação.



3




Regras

A normalização é um conjunto de regras que devem ser seguidas. São as chamadas **formas normais**.

As bibliografias apontam a obrigatoriedade de 3 formas normais. Existem outras 3, mas estão caindo em desuso, pois considera-se que as 3 obrigatórias atendam a todas as necessidades.

Vamos estudar as 3 formas normais obrigatórias. Elas são identificadas como:

- 1FN - Primeira Forma Normal
- 2FN - Segunda Forma Normal
- 3FN - Terceira Forma Normal



4

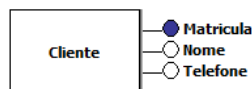
1FN – Primeira Forma Normal

Diz-se que uma tabela está na 1FN se e somente se:

- Os atributos são atômicos.

A regra é simples. Em cada atributo só deve existir cadastrado um valor.

Exemplo de tabela que **não** está na 1FN:



Não enxergou o problema? Então observe um exemplo de cadastro feito nesta tabela:

Cliente

Matrícula	Nome	Telefone
1	João Miguel	(21) 98871-5612
2	Rafael Souza	(21) 2233-1121, (21) 97333-0041
3	Aline Reis	(22) 96624-6621

Temos mais de um número de telefone cadastrado no mesmo atributo



5

FIRJAN



INFORMA, FORMA, TRANSFORMA.

2FN – Segunda Forma Normal

Uma tabela está na segunda forma normal se:

- Estiver na 1FN
- Ao possuir uma chave primária composta, todos os atributos comuns (não pertencentes às colunas da chave primária composta) são dependentes de todos os atributos que compõem a chave primária composta.

A 2FN é uma regra em que a sua aplicação só se faz necessária caso alguma tabela possua uma **chave primária composta** (por mais de uma coluna). Tabelas que possuem chave primária simples e estão na 1FN automaticamente já estão também na 2FN.

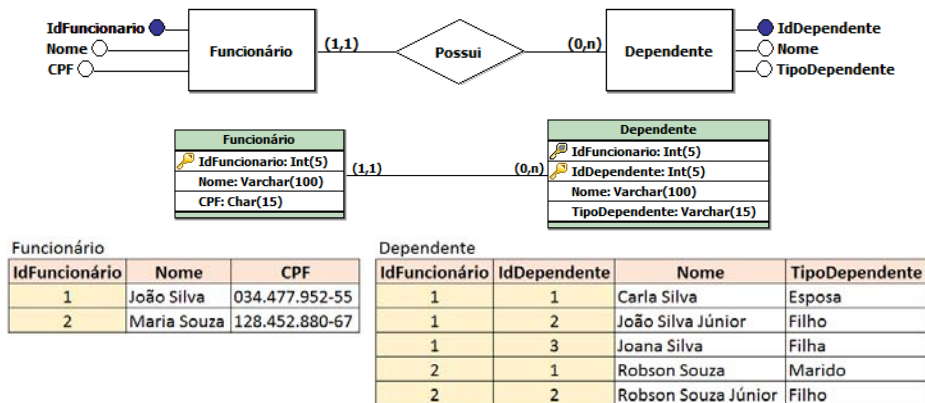
As tabelas que possuem chave primária composta precisam sofrer a verificação da 2FN, pois não podem possuir colunas fora da chave primária composta que dependam parcialmente da chave primária, ou seja, dependam apenas de um campo da chave primária composta.



6

2FN – Segunda Forma Normal

Observe a modelagem e exemplo de preenchimento de tabelas onde existe uma chave primária composta:



Na tabela *Dependente* é possível identificar facilmente de quem cada pessoa é dependente graças à chave primária composta.



7

2FN – Segunda Forma Normal

Funcionário			Dependente			
IdFuncionario	Nome	CPF	IdFuncionario	IdDependente	Nome	TipoDependente
1	João Silva	034.477.952-55	1	1	Carla Silva	Esposa
2	Maria Souza	128.452.880-67	1	2	João Silva Júnior	Filho
			1	3	Joana Silva	Filha
			2	1	Robson Souza	Marido
			2	2	Robson Souza Júnior	Filho

Estas tabelas **estão na 2FN** pelos seguintes motivos:

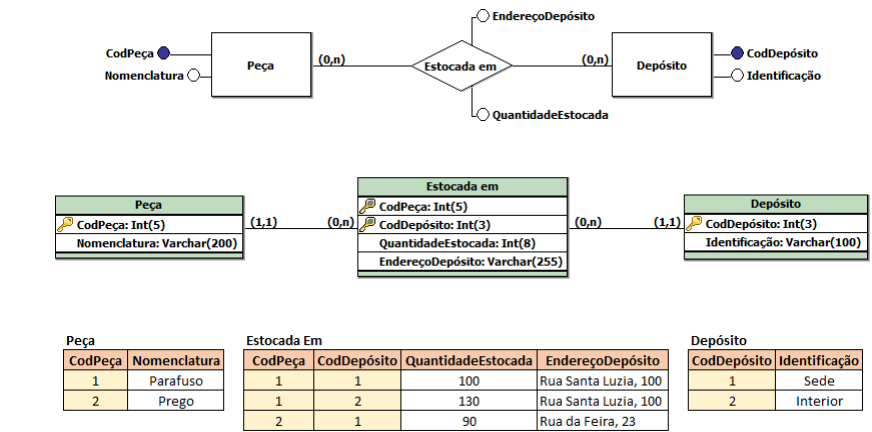
- Ambas estão na 1FN
- A tabela *Funcionário* não possui chave primária composta, logo ela já está automaticamente na 2FN
- A tabela *Dependente* possui chave primária composta e cada dado cadastrado em um atributo não chave (*Nome* e *TipoDependente*) depende totalmente dos 2 valores existentes na chave primária composta (*IdFuncionario* e *IdDependente*).



8

2FN – Segunda Forma Normal

Agora observe esta outra modelagem e exemplo de tabelas preenchidas:



9

2FN – Segunda Forma Normal

Peça		Estocada Em				Depósito	
CodPeça	Nomenclatura	CodPeça	CodDepósito	QuantidadeEstocada	EndereçoDepósito	CodDepósito	Identificação
1	Parafuso	1	1	100	Rua Santa Luzia, 100	1	Sede
2	Prego	1	2	130	Rua Santa Luzia, 100	2	Interior
		2	1	90	Rua da Feira, 23		

A tabela *EstocadaEm* **não está na 2FN**.

Percebe-se que o atributo *EndereçoDepósito* depende somente de uma das colunas da chave primária composta (*CodDepósito*).

Isto quer dizer que *EndereçoDepósito* é algo que diz respeito apenas a *CodDepósito*. Este atributo não deveria estar na tabela *EstocadaEm*, e sim na tabela *Depósito*.

10

3FN – Terceira Forma Normal

Uma tabela está na terceira forma normal se:


- Estiver na 2FN
- Não existir dependência entre seus atributos não-chave

Ou seja, uma coluna que não seja chave primária não pode depender de outra coluna que também não seja chave primária.

Entre os erros mais comuns que são barrados pela 3FN estão:

- Coluna comum que armazena algum tipo de cálculo sobre outra coluna comum.
- Coluna comum que explica algo de outra coluna comum.

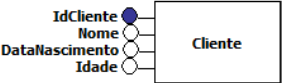
Vamos observar alguns exemplos a seguir:



11

3FN – Terceira Forma Normal

Exemplo 1:




Cliente			
IdCliente: Int(6)			
Nome: Varchar(100)			
DataNascimento: Date			
Idade: Int(3)			

Cliente			
IdCliente	Nome	DataNascimento	Idade
1	Aline Silva	2000/04/18	15
2	João Alfredo	1990/02/17	25
3	Márcia Baptista	1995/08/01	19

ERRO: *Idade* depende de *DataNascimento*, e pode ser calculada ao invés de ser cadastrada.

Existem funções SQL que conseguem calcular a idade do cliente a partir da data de nascimento cadastrada.

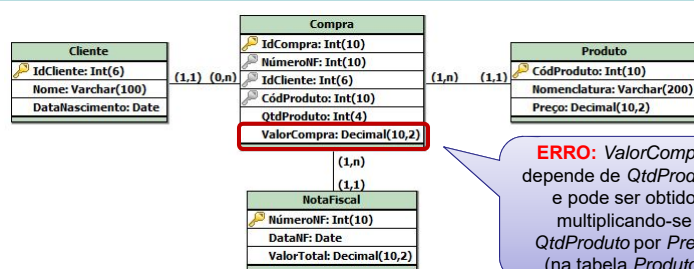
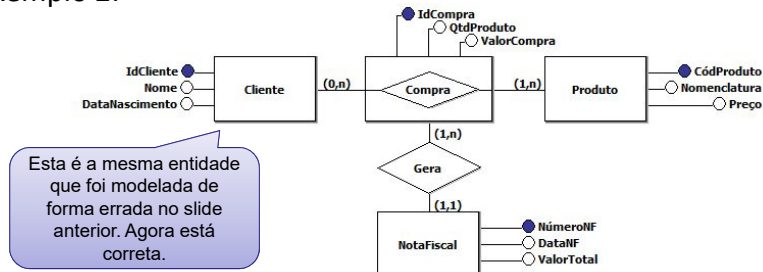
Da forma que está modelado neste exemplo o campo *Idade* precisará ser atualizado manualmente pelo usuário do SGBD, o que certamente não será feito de forma correta, ainda mais se a tabela possuir muitos clientes cadastrados.



12

3FN – Terceira Forma Normal

Exemplo 2:



13

Soluções

Sempre que uma falha é detectada ao ser aplicada alguma regra de normalização, a solução cai no replanejamento da modelagem.

O que costuma ser a solução do problema é:

1FN - A falha detectada é algum campo contendo mais de um valor cadastrado. Solução: Geração de nova tabela e relacionamento com a que deu origem ao problema.

2FN – A falha detectada é algum campo que se relaciona apenas com um dos componentes da chave primária composta. Solução: Este campo pertence a outra tabela, ou ainda, a mesma solução descrita acima.

3FN – A falha detectada é algum campo comum que se refere a outro campo comum. Solução: Este campo é desnecessário e deve ser apagado, ou ainda, a mesma solução da 2FN.

14

Estudo de Caso

Uma abordagem inicial sobre uma escola tratando a questão aluno x turma:

Matricula

Nome

Identidade

CPF

DataNascimento

Idade

Telefone

Endereço

Aluno

Curso

DataInicio

DataFim

Turma


Modelo Conceitual Desnormalizado

Exemplo de preenchimento desta tabela no Banco de Dados

Aluno											
Matricula	Nome	Identidade	CPF	Data Nascimento	Idade	Telefone	Endereço	Curso	DataInicio	DataFim	Turma
1	João Santos	100667658	21558994745	1995/05/15	19	3344-3322	R. Feliz, 18, Leblon, Rio de Janeiro	Inglês	03/08/2012	08/04/2013	2012034
2	Maria Silva	155883320	25880116555	1999/09/27	15	2233-2323	R. Conceição, 79 AP104, Lapa, Rio	Francês	18/10/2013	20/01/2013	2013181
3	Alan Souza	855442586	87554775800	2003/01/03	12	2134-9876	Rua da Feira, 345, apart.103, Bangu, Rj	ingles	03/08/2012	08/06/2013	2012034

Observando a tabela acima é possível perceber algumas falhas:

- *Redundância 1:* 2012034 foi escrito 2 vezes.
- *Redundância 2:* Inglês também.
- *Inconsistência 1:* O curso é de *Inglês* ou *ingles* ?
- *Inconsistência 2:* Qual a data de fim de curso da turma 2012034 ?
- *Inconsistência 3:* Afinal, como deve ser registrado Rio de Janeiro no endereço?
- *Inconsistência 4:* E como deve ser registrado um apartamento no endereço?
- *Inconsistência 5:* O usuário vai mesmo calcular a idade de cada aluno?



15

Estudo de Caso

Uma abordagem inicial sobre uma escola tratando a questão aluno x turma:

Matricula

Nome

Identidade

CPF

DataNascimento

Idade

Telefone

Endereço

Aluno

Curso

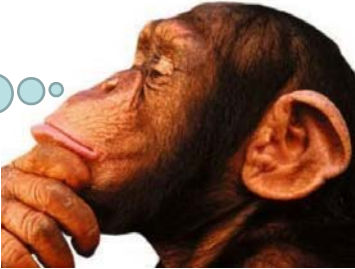
DataInicio


DataFim

Turma

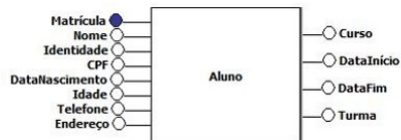
Aluno											
Matricula	Nome	Identidade	CPF	Data Nascimento	Idade	Telefone	Endereço	Curso	DataInicio	DataFim	Turma
1	João Santos	100667658	21558994745	1995/05/15	19	3344-3322	R. Feliz, 18, Leblon, Rio de Janeiro	Inglês	03/08/2012	08/04/2013	2012034
2	Maria Silva	155883320	25880116555	1999/09/27	15	2233-2323	R. Conceição, 79 AP104, Lapa, Rio	Francês	18/10/2013	20/01/2013	2013181
3	Alan Souza	855442586	87554775800	2003/01/03	12	2134-9876	Rua da Feira, 345, apart.103, Bangu, Rj	ingles	03/08/2012	08/06/2013	2012034

Caramba, foram só 3 linhas!
Imagine se cada turma
possuir 40 alunos?





Vamos normalizar ?



Passo1: Diagrama Original

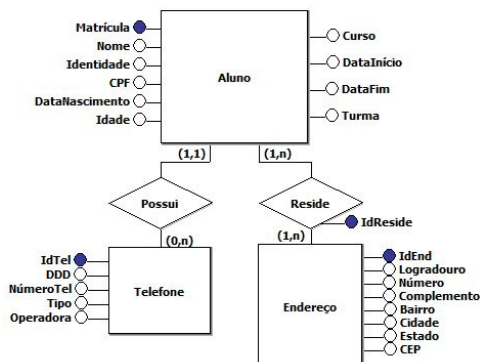
Observar o diagrama inicial em busca de falhas.

Comece buscando algo que desrespeite a 1FN.



17

Passo2: Aplicando a 1FN



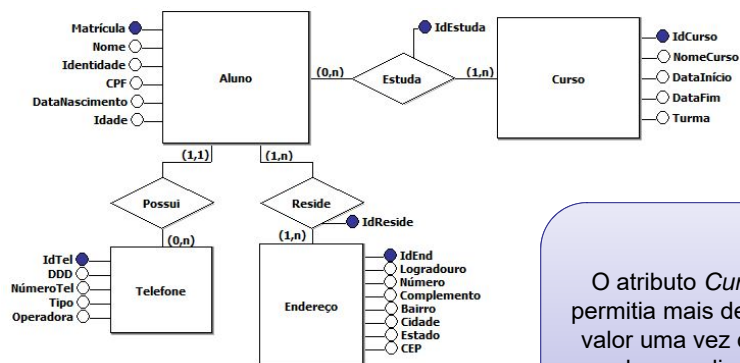
Atributo *Telefone* permitia o cadastro de mais de um número de telefone para o mesmo aluno.

Atributo *Endereço* por si só já exige diversas informações num mesmo campo. Basta observar em quantos campos ele foi desmembrado .



18

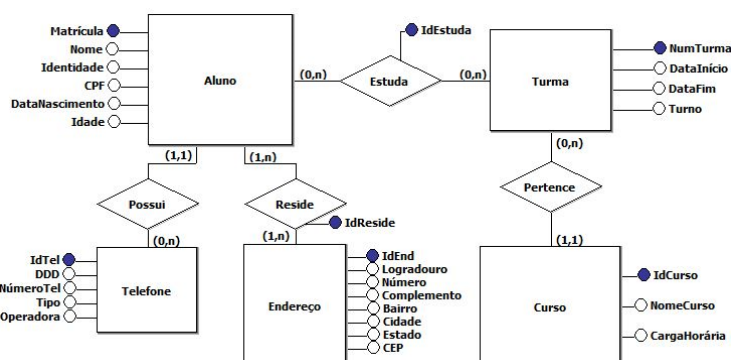
Passo3: Aplicando mais uma vez a 1FN:



O atributo *Curso* permitia mais de um valor uma vez que um aluno realizasse matrícula em mais de um curso.

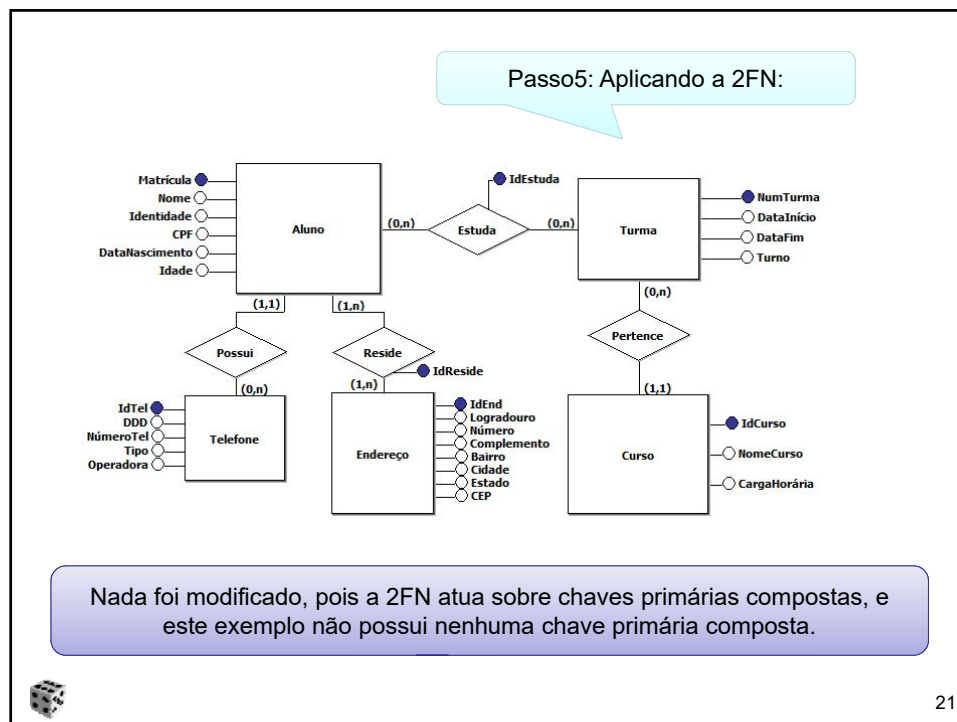
19

Passo4: Aplicando novamente a 1FN:

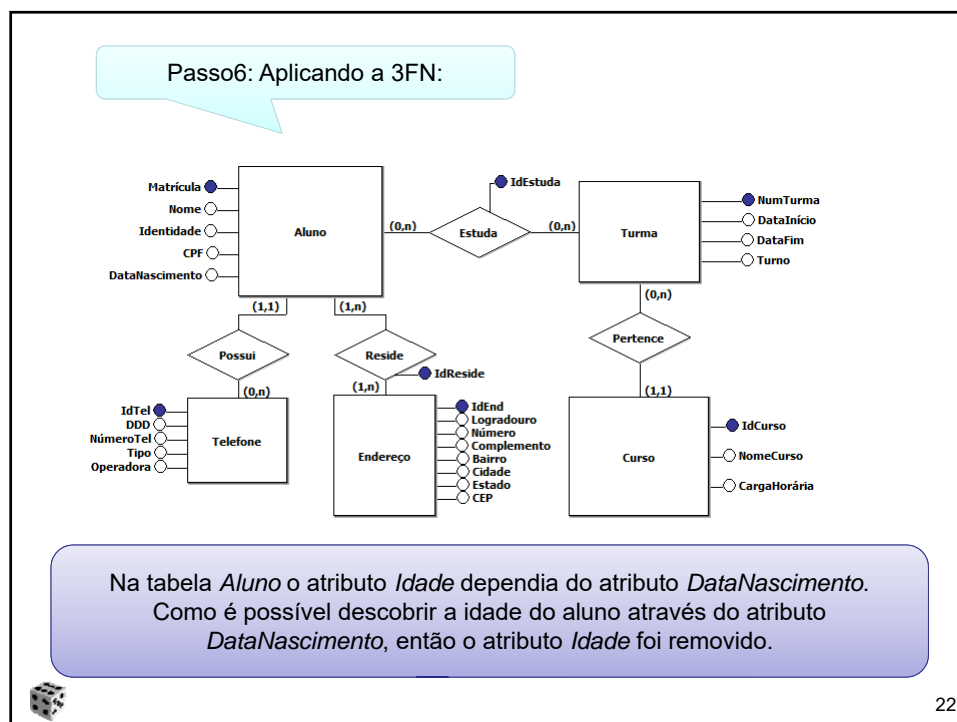


É possível que existam diversas turmas de um mesmo curso, logo o atributo *Turma* precisou ser convertido em uma entidade relacionada à *Curso*.

20



21



22



The slide has a blue header with the FIRJAN logo and the text 'INFORMA, FORMA, TRANSFORMA.'. The main content area is white and contains the title 'Bibliografia' in a large black font. Below the title are three book covers, each followed by its title, author, and publisher.

	<p>Introdução ao Design de Bancos de Dados Clare Churcher Ed. Altabooks</p>
	<p>Modelagem Conceitual e Projeto de Bancos de Dados Paulo Cougo Ed. Campus</p>
	<p>Modelagem Lógica de Dados: Construção Básica e Simplificada Eduardo Bernardes Castro Ed. Ciência Moderna</p>

24