

Objetivos:

- I. Normalização;
- II. Primeira forma normal;
- III. Segunda forma normal;
- IV. Terceira forma normal;
- V. O processo de modelagem de dados.

I. Normalização

É o processo pelo qual um esquema de relação (tabela) insatisfatório é quebrado de forma que seus atributos formem relações menores que sejam mais adequadas:

- Sem redundância de dados;
- Maior facilidade de manutenção.

O impacto da normalização será sentido nas operações no SGBD.

Diz-se que uma tabela num BD relacional está numa certa “forma normal” se satisfaz certas condições. Aqui apresentaremos três formas de normalização:

- Primeira forma normal (1FN);
- Segunda forma normal (2FN);
- Terceira forma normal (3FN).

Apesar de existirem cinco formas normais, considera-se que as bases de dados estão normalizadas se satisfizerem a 3FN.

II. Primeira forma normal

Um esquema de relação está na 1FN se, e somente se, todos os valores nas células são atômicos, o que significa que as células da tabela não podem ter mais de um valor. Por exemplo, atributos multivalorados e compostos não são atômicos.

Cliente

Cpf	Nome	Telefones
123456789	Maria Silva	1234567890; 12987654321
234567890	Paulo Alves	1291234567

O atributo **Telefones** é multivalorado, veja que esta célula possui mais de um conteúdo. Esse tipo de situação dificulta o processo de busca, exclusão e alteração de telefone

Para satisfazer a 1FN precisamos criar uma tabela e colocar os telefones nela. Aqui criamos a tabela **Telefone** e usamos o **Cpf** como chave estrangeira para ligar as duas tabelas.

Cliente		Telefone		
Cpf	Nome	Código	Cpf	Número
123456789	Maria Silva	1	123456789	1234567890
234567890	Paulo Alves	2	123456789	12987654321
		3	234567890	1291234567

↑ Chave primária

↑ Chave primária ↑ Chave estrangeira

No exemplo a seguir, o atributo **Endereço** é composto, veja que ele é formado por mais de um valor. Esse tipo de situação dificulta o processo de busca, exclusão e alteração de endereço.

Cliente		
Cpf	Nome	Endereço
123456789	Maria Silva	R. Um, 123, Vila São João
234567890	Paulo Alves	Av. Dois, 321, Parque do Bosque

Para satisfazer a 1FN precisamos separar os atributos compostos em colunas. Cada célula resultante possui um valor atômico.

Cliente				
Cpf	Nome	Logradouro	Número	Bairro
123456789	Maria Silva	R. Um	123	Vila São João
234567890	Paulo Alves	Av. Dois	321	Parque do Bosque

III. Segunda forma normal

Um esquema de relação está na 1FN se, e somente se, todos os valores nas células são atômicos, o que significa que as células da tabela não podem ter mais de um valor. Por exemplo, atributos multivalorados e compostos não são atômicos.

Um esquema de relação está na 2FN se, e somente se, estiver na 1FN e cada atributo não-chave for dependente funcional da chave primária inteira, ou seja, cada atributo não-chave não poderá ser dependente funcional de apenas parte da chave primária.

Ocorre somente quando a chave primária é composta por mais de um atributo.

No exemplo a seguir a chave primária da tabela **Empregado** é composta pelos atributos **Cpf** e **NroProjeto**.

Empregado

Cpf	NroProjeto	NomeEmpregado	Horas	NomeProjeto	LocalProj
123456789	1	Maria Silva	20	Raio X	Produção
234567890	2	Paulo Alves	30	Melhorias	RH
567890123	2	Mariana Souza	40	Melhorias	RH

↑ ↑
Chave primária Chave primária

A relação **Empregado** está na 1FN mas não está na 2FN, porque o atributo **NomeEmpregado** viola a 2FN pelo fato dele possuir a dependência funcional do atributo-chave **Cpf**, mas não depende funcionalmente do atributo-chave **NroProjeto**.

O mesmo ocorre com os atributos **NomeProjeto** e **LocalProj**, que dependem funcionalmente do atributo-chave **NroProjeto**, mas não dependem do atributo-chave **Cpf**.

Para normalizar a relação **Empregado** na 2FN, é necessário decompor essa relação em relações na 2FN, onde os atributos não-chave estarão associados somente com a parte da chave-primária que eles dependem funcionalmente.

As dependências funcionais **Cpf → NomeEmpregado**, **NroProjeto → NomeProjeto** e **NroProjeto → LocalProj** levam à decomposição da relação **Empregado** em três relações, assim como é mostrado a seguir.

Empregado

Cpf	NomeEmpregado
123456789	Maria Silva
234567890	Paulo Alves
567890123	Mariana Souza

↑
Chave primária

Projeto

NroProjeto	NomeProjeto	LocalProj
1	Raio X	Produção
2	Melhorias	RH

↑
Chave primária

Hora_trabalho_por_projeto

Cpf	NroProjeto	Horas
123456789	1	20
234567890	2	30
567890123	2	40

↑
Chave primária e
chave estrangeira da
tabela Empregado

↑
Chave primária e
chave estrangeira da
tabela Projeto

A relação **Hora_trabalho_por_projeto** se tornou a quebra do relacionamento N:N entre as relações **Empregado** e **Projeto**.

IV. Terceira forma normal

Uma relação está na 3FN se, e somente se, estiver na 2FN e não tiver dependências transitivas.

A dependência transitiva ocorre quando um atributo não-chave, além de depender da chave primária, depende funcionalmente de outro atributo ou combinação de atributos não-chave.

Em uma relação na 3FN não existem atributos não-chave que dependem funcionalmente de outros atributos não-chave.

A relação Empregado está na 2FN, mas não está na 3FN, porque o atributo **NroDep** não é chave primária e os atributos **NomeDep** e **Gerente** dependem dele para serem identificados. Isso causa dependência transitiva.

Empregado

Cpf	NomeEmpregado	DataNasc	NroDep	NomeDep	Gerente
123456789	Maria Silva	10/11/1981	1	RH	123456789
234567890	Paulo Alves	01/02/1982	2	Produção	234567890
567890123	Mariana Souza	02/03/1993	2	Produção	234567890

↑
Chave primária

↑ ↑
Dependem do atributo NroDep
para serem identificados

A relação Empregado pode ser normalizada para a 3FN pela sua decomposição em duas relações que atendam a 3FN, assim como é mostrado a seguir.

Empregado

Cpf	NomeEmpregado	DataNasc	NroDep
123456789	Maria Silva	10/11/1981	1
234567890	Paulo Alves	01/02/1982	2
567890123	Mariana Souza	02/03/1993	2

↑
Chave primária

↑
Chave estrangeira

Departamento

NroDep	NomeDep	Gerente
1	RH	123456789
2	Produção	234567890

↑
Chave primária

A decomposição para satisfazer a 3FN consiste em dividir a relação de modo que, os atributos sejam agrupados de acordo com as suas dependências funcionais, incluindo as transitivas. Como resultado, nas relações Empregado e Departamento todos os atributos não-chave dependem de forma não transitiva da chave-primária.

V. O processo de modelagem de dados

A modelagem consiste em distribuir os dados em tabelas de forma a evitar redundâncias e sobreposições.

Problema 1 - Centralizar os dados numa única tabela

No exemplo a seguir os atributos da fazenda foram colocados numa única tabela. Esse processo facilita a consulta, mas os dados do município está repetido em vários registros. Se no processo de inserir um novo registro o usuário digitar o nome do município errado, então teremos inconformidade nos dados. Além de ser necessário alterar vários registros para fazer a correção no nome de um município.

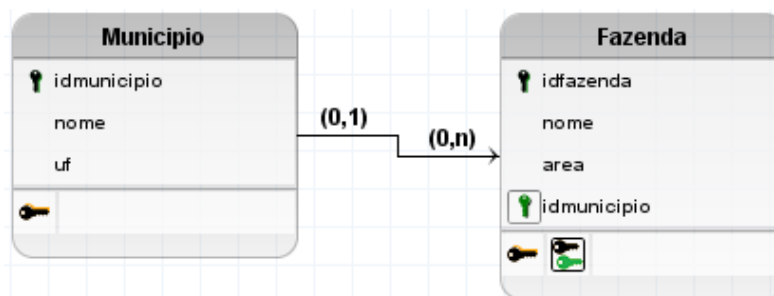
Outra falha neste modelo, é a falta de uma chave primária forte, pois podem existir várias fazendas com o mesmo nome no mesmo município.



nomefazenda	areafazenda	municipio	uf
Água Clara	45.7	Piracaia	SP
Borda da Mata	54.2	Piracaia	SP
Capim Branco	105.8	São Bento do Sapucaí	MG
Curralinho	280.2	Monteiro Lobato	SP
Matão	185.4	Camanducaia	MG
Monte Alto	90.8	São Bento do Sapucaí	MG
Monte Verde	18.1	São Bento do Sapucaí	MG
Pinheiro	90.5	Monteiro Lobato	SP
Pouso Alto	17.9	Brasópolis	MG
Três Irmãos	9.3	Camanducaia	MG

O correto é agrupar os atributos por característica em duas tabelas e fazer a ligação entre elas usando a chave estrangeira. O relacionamento entre Municipio e Fazenda é 1:n, onde um município pode ter várias fazendas, mas uma fazenda só pode estar em um município. A chave primária do lado 1 vai para o lado n como chave estrangeira, por este motivo o atributo idmunicipio foi colocado na Fazenda.

Veja que os dados de um município são cadastrados apenas uma vez na Municipio, o que repete na Fazenda é o idmunicipio. Esse processo tem a vantagem de centralizar os dados do município.



Municipio

idmunicipio	nome	uf
1	Brasópolis	MG
2	Paraisópolis	MG
3	São Bento do Sapucaí	MG
4	Camanducaia	MG
5	Piracaia	SP
6	Monteiro Lobato	SP

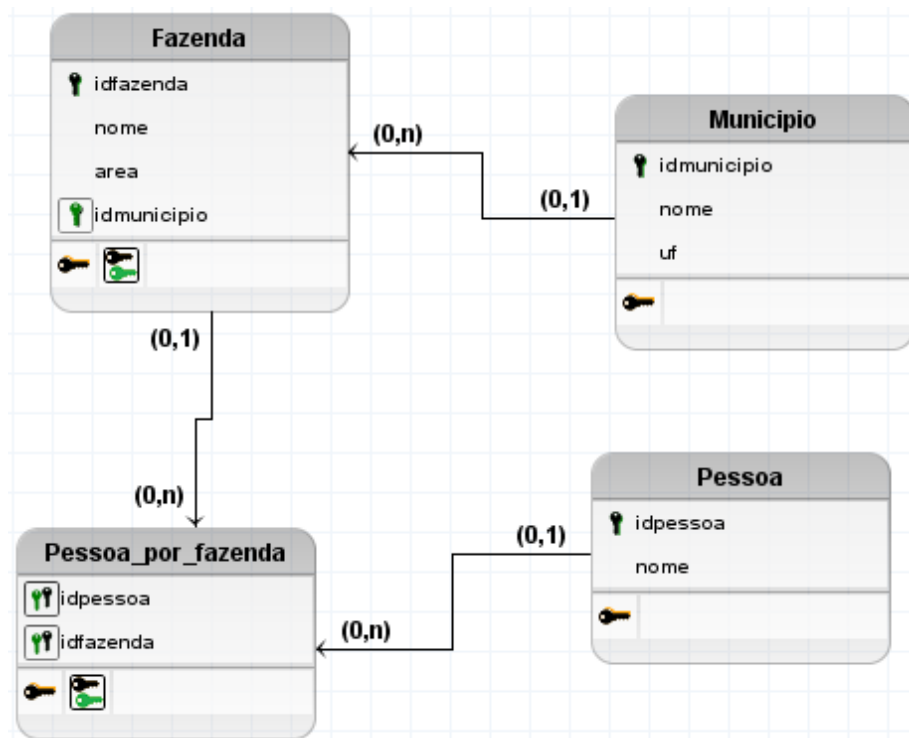
Fazenda

idfazenda	idmunicipio	nome	area
1	4	Matão	185.4
2	6	Curralinho	280.2
3	3	Monte Alto	90.8
4	1	Pouso Alto	17.9
5	5	Borda da Mata	54.2
6	3	Capim Branco	105.8
7	4	Três Irmãos	9.3
8	5	Água Clara	45.7
9	3	Monte Verde	18.1
10	6	Pinheiro	90.5

Problema 2 – Relacionamento N:N (muitos para muitos)

Considere que uma fazenda pode ter vários donos e uma pessoa pode ser proprietária de várias fazendas. Neste caso existe um relacionamento N:N entre as entidades fazenda e pessoa.

Quando existe um relacionamento N:N cria-se uma tabela no meio para receber as chaves estrangeiras. No diagrama a seguir a tabela Pessoa_por_fazenda foi criada para receber as chaves estrangeiras, desta forma, para criar uma relação de propriedade, insere-se um registro na Pessoa_por_fazenda.



Pessoa_por_fazenda

idpessoa	idfazenda
1	1
8	1
4	2
1	3
6	3
6	4
2	5
3	6
5	7
3	8
7	8
4	8
2	9
1	10

Pessoa

idpessoa	nome
1	Bruno Almeida
2	Carla Prado
3	Rodrigo Taveres
4	Antonio Silva
5	Stephanie Lara
6	Simone Braga
7	Aline Almeida
8	João Bosco

Exercícios

Exercício 1: Considere a necessidade de criar um BD para armazenar um cadastro de pessoas com os seguintes atributos: nome, e-mail e telefone. Considere que:

- Uma pessoa pode ter vários telefones;
- Uma pessoa pode ter vários e-mails.

Para checar se o seu modelo lógico está correto copie os registros a seguir para as tabelas que você criou no seu modelo. O seu resultado estará correto se os nomes, telefones e e-mails não se repetirem, ou seja, os valores só podem estar em uma única célula, e todas as células precisam ter valores atômicos.

Nome	Mail	Número
Ana	ana@teste.com.br	12988775544
Ana	ana@teste.com	12988775544
Pedro	pedro@teste.com	12977556633
Pedro	pedro@teste.com	1234567890
Maria	maria@teste.com.br	12966554433
Maria	maria@teste.com	12966554433
Maria	maria@teste.com.br	123345566
Maria	maria@teste.com	123345566

Exercício 2: Considere a necessidade de criar um BD para armazenar a agenda de atividades dos usuários. Considere que:

- Um usuário pode ter várias atividades;
- Cada atividade está associada a um endereço;
- Cada atividade possui uma data e horário;
- Cada atividade possui uma observação, por exemplo, levar máscara.

Utilize os dados a seguir para validar o seu modelo lógico.

Nome	Atividade	Observação	Data	Localidade	Endereço
Ana	Médico	Levar máscara	01/02/2022 14:30	Clínica Rio Sul	R. Senador Viana, 123, Vila Santo Antônio, Jacareí, SP
Pedro	Dentista	Pedir guia do raio X	05/02/2022 09:15	Centro Odontológico	Av. Maria Antonieta, 81, Parque Rio Comprido, Jacareí, SP
Maria	Médico	Levar receitas antigas	08/02/2022 17:45	Clínica Rio Sul	R. Senador Viana, 123, Vila Santo Antônio, Jacareí, SP
Pedro	Médico	Levar resultado dos exames	09/02/2022 13:50	Clínica Rio Sul	R. Senador Viana, 123, Vila Santo Antônio, Jacareí, SP
Maria	Dentista	Evitar frutas cítricas	15/02/2022 14:30	Centro Odontológico	Av. Maria Antonieta, 81, Parque Rio Comprido, Jacareí, SP

Exercício 3: Considere a necessidade de criar um BD para armazenar a localização dos motoboys em intervalos de tempo. Considere que:

- Um motoboy possui nome, placa da moto e número do Whatsapp;

- A localização do motoboy é formada pelas coordenadas latitude e longitude e o horário.

Utilize os dados a seguir para validar o seu modelo lógico.

Nome	Atividade	Telefone	Horario	Latitude	Longitude
Lucas	ABC1234	12988775566	01/02/2022 08:00	-23.317	-45.982
Lucas	ABC1234	12988775566	01/02/2022 08:10	-23.314	-45.979
Lucas	ABC1234	12988775566	01/02/2022 08:20	-23.302	-45.970
Otávio	QWE8901	12900118877	01/02/2022 14:00	-23.296	-45.947
Otávio	QWE8901	12900118877	01/02/2022 14:10	-23.276	-45.939