

Primeira Forma Normal

Uma tabela 1FN deve estar de acordo com as seguintes informações:

- os dados devem estar em valores atômicos, ou seja, indivisíveis;
- ter apenas um dado por coluna;
- deve existir pelo menos uma chave primária;
- se existirem atributos multivalorados, uma nova tabela será criada;
- se existirem atributos compostos, eles serão desmembrados em novas colunas de dados.

Exemplo da 1FN

Logradouro	Número	Cidade	UF	País
Rua Lima	42	São Paulo	SP	Brasil
Rua do Barbalho	56	Recife	PE	Brasil
Avenida Brasil	123	São Paulo	SP	Brasil

Nos atributos **compostos** ocorre a decomposição dos dados, em que cada dado é adicionado em uma coluna. Assim, o atributo **endereço** será subdividido em atributos menores.

Exemplo da 1FN

Pessoa			
CPF	Nome	Endereço	Contato
123.456.789-10	Maria Dias	Rua Lima, 42, São Paulo, SP, Brasil	21 1234-5678
123.456.789-20	Carlos Monteiro	Rua do Barbalho, 56, Recife, PE, Brasil	35 0125-8745 35 8930-4567
123.456.789-30	Kauan Queiroz	Avenida Brasil, 123, São Paulo, SP, Brasil	41 5644-2310 41 0011-2233

Para que essa tabela esteja na 1FN, precisamos organizar alguns dados. Percebemos que nela existem atributos multivalorados, que são aqueles que podem ter mais de um valor, como o **Contato**, e atributos compostos, que são aqueles que podem ser divididos em atributos menores, como o **Endereço**. Então, é preciso modificar isso, pois não é permitido nesse tipo de forma normal.

Exemplo da 1FN

Contato		
Identificador	CPF	Contato
1	123.456.789-10	21 1234-5678
2	123.456.789-20	35 0125-8745
3	123.456.789-20	35 8930-4567
4	123.456.789-30	41 5644-2310
5	123.456.789-30	41 0011-2233

No caso dos atributos **multivalorados**, criamos uma nova tabela com os dados. Na nova tabela **Contato**, a chave primária é o identificador e a chave estrangeira é o CPF correspondente da tabela **Pessoa**. Se um CPF possuir dois contatos, eles devem ser apresentados em linhas diferentes na tabela Contato, como é o caso dos CPFs dos identificadores 4 e 5.

Exemplo da 2FN

Identificador	Identificador_função	Nome	Função	Descrição	Data_inicio
1	103	Carla	Programadora	Desenvolvedora Python	08/03/2020
2	87	Rian	Designer	Comunicação visual	12/03/2021
3	104	Carla	Programadora	Desenvolvedora Java	08/03/2020
4	01	Camila	Gerente	Planejamento	18/11/2021
5	54	Pedro	Analista	Análise resultados	01/04/2022
6	54	Camila	Gerente	Análise resultados	01/04/2022

Essa tabela está na 1FN, não possuindo dados multivalorados e nem atributos compostos. Portanto, como não está na 2FN, ela precisa de ajustes.

Nesta tabela, temos chaves primárias compostas, o **IDENTIFICADOR** e o **IDENTIFICADOR_FUNÇÃO**. Com o atributo NOME, conseguimos saber qual é o nome da pessoa, sendo um atributo que depende parcialmente da chave primária.



Exemplo da 2FN

Identificador	Identificador_função	Nome	Função	Descrição	Data_inicio
1	103	Carla	Programadora	Desenvolvedora Python	08/03/2020
2	87	Rian	Designer	Comunicação visual	12/03/2021
3	104	Carla	Programadora	Desenvolvedora Java	08/03/2020
4	01	Camila	Gerente	Planejamento	18/11/2021
5	54	Pedro	Analista	Análise resultados	01/04/2022
6	54	Camila	Gerente	Análise resultados	01/04/2022

Para construir a outra tabela, é interessante montar o modelo conceitual para entendermos quais atributos ficarão em cada uma das tabelas.

Exemplo da 2FN

Identificador	Identificador_função	Nome	Função	Descrição	Data_inicio
1	103	Carla	Programadora	Desenvolvedora Python	08/03/2020
2	87	Rian	Designer	Comunicação visual	12/03/2021
3	104	Carla	Programadora	Desenvolvedora Java	08/03/2020
4	01	Camila	Gerente	Planejamento	18/11/2021
5	54	Pedro	Analista	Análise resultados	01/04/2022
6	54	Camila	Gerente	Análise resultados	01/04/2022

Também conseguimos saber qual é a função da pessoa pelo número do identificador e a descrição pelo identificador da função. Com isso, podemos ter duas tabelas ao invés de apenas uma.

Para transformar essa tabela na 2FN, será necessário desmembrar alguns dados e construir outra tabela. Assim, nenhuma coluna depende parcialmente de outra.

Segunda Forma Normal

A **Segunda Forma Normal (2FN)** tem que estar, obrigatoriamente, na 1FN. Sendo assim, não deve ter atributos compostos ou multivalorados e nem dependências parciais.

No caso das **chaves primárias compostas**, a 2FN exige que todas as relações tenham dependência total. Ou seja, uma chave primária composta determina funcionalmente os atributos, não podendo existir um atributo que depende apenas de uma parte dessa chave.

Exemplo da 2FN



Para que a tabela original fique na 2FN, ela precisa ser dividida em duas. Com esse diagrama, temos a organização de como as duas tabelas seriam criadas com os seus devidos atributos para que não haja chave primária composta. No relacionamento das entidades, uma pessoa pode possuir uma ou várias funções e uma função pode ter uma ou várias pessoas envolvidas.

Terceira Forma Normal

Na 3FN, os dados:

- devem estar na 2FN;
- Os atributos não podem depender de outros que não são chaves.

Exemplo da 3FN

Vendedor	
Identificador_vendedor	Nome_vendedor
001	Carol
010	Ruan
100	José
001	Carol

Agora, temos uma tabela Vendedor com a sua chave primária **Identificador_vendedor** e os nomes de cada vendedor.

Exemplo da 3FN

Nota_fiscal	Identificador_vendedor	Nome_vendedor	Codigo_Produto	Quantidade_vendido
12345	001	Carol	0123	200
65789	010	Ruan	9012	50
32456	100	José	3210	120
87689	001	Carol	5643	30

Essa tabela não está na 3FN, pois os seus atributos não são todos dependentes. A **Nota_fiscal** é a chave primária e sabemos o **Nome_vendedor** através do **Identificador_vendedor**. Com isso, temos uma dependência.

Para resolver isso, precisamos criar uma nova tabela com esses atributos que são dependentes.

Exemplo da 3FN

Nota_fiscal	Identificador_vendedor	Nome_vendedor	Codigo_Produto	Quantidade_vendido
12345	001	Carol	0123	200
65789	010	Ruan	9012	50
32456	100	José	3210	120
87689	001	Carol	5643	30

Os nomes dos vendedores são excluídos desta tabela, porém ela continua com o **Identificador_vendedor**, que se torna uma chave estrangeira nesta tabela, pois esse atributo é uma chave primária na tabela Vendedor.

Forma Normal de Boyce-Codd (FNBC)

Cada forma normal é estritamente mais forte do que a anterior, então:

- toda relação da 2FN encontra-se na 1FN; e
- toda relação na 3FN encontra-se na 2FN.

O mesmo acontece com a Forma Normal de Boyce-Codd (FNBC). Toda relação FNBC está na 3FN.

Forma Normal de Boyce-Codd (FNBC)

Ela é uma melhoria da 3FN utilizada em casos onde a tabela apresenta mais de uma chave candidata. A FNBC serve para separar, em novas tabelas, valores que ainda estejam redundantes do seu banco de dados.

Forma Normal de Boyce-Codd (FNBC)

Escola		
Aluno	Disciplina	Professor
Mel	Introdução a Programação	Rafael
Pietro	Programação Orientada a Objetos	Cintia
Carlos	Teoria da Computação	Queiroz
Rita	Introdução a Programação	Maria
Mel	Programação Orientada a Objetos	Tiago
Carlos	Introdução a Programação	Rafael

Analisando essa tabela, conseguimos entender que:

- descobrimos o professor se soubermos quem é o aluno e qual é a disciplina;
- descobrimos qual é a disciplina se soubermos quem é o professor;

Assim, existem dependentes entre os atributos.

Para transformar essa tabela na FNBC, é necessário decompor as relações que são dependentes. Neste caso, essa tabela é decomposta da seguinte forma:

- Professor, Disciplina;
- Aluno, Professor.

Forma Normal de Boyce-Codd (FNBC)

Aluno:PK	Professor:FK
Mel	Rafael
Pietro	Cintia
Carlos	Queiroz
Rita	Maria
Mel	Tiago
Carlos	Rafael

Após a decomposição, temos outra tabela com a chave primária **Aluno** e **Professor** como chave estrangeira, pois, a partir do professor, podemos saber qual disciplina o aluno irá cursar.



Forma Normal de Boyce-Codd (FNBC)

Vimos no exemplo anterior como transformar uma tabela para a FNBC, utilizando a função de decompor os atributos, construindo novas tabelas para eliminar as dependências.

Forma Normal de Boyce-Codd (FNBC)

Professor:PK	Disciplina
Rafael	Introdução a Programação
Cintia	Programação Orientada a Objetos
Queiroz	Teoria da Computação
Maria	Introdução a Programação
Tiago	Programação Orientada a Objetos
Rafael	Introdução a Programação

Após a decomposição, temos uma tabela com a chave primária **Professor** e sua respectiva **Disciplina**.

