

RELATÓRIO DE IDENTIFICAÇÃO E AJUSTE DAS TABELAS PARA ATIVIDADE UC4

Aluno: André Cararo Lowcke

Atividade

Observando as tabelas citadas no contexto e realizando as consultas no banco de dados, elabore um relatório respondendo aos seguintes pontos:

1. Por que a tabela "Cliente" não está na 1FN? Proponha uma solução para fazer sua normalização na 1FN.
2. Por que a tabela "Item_venda" não está na 2FN? Proponha uma solução para fazer sua normalização na 2FN.
3. Por que a tabela "venda" não está na 3FN? Proponha uma solução para fazer sua normalização na 3FN.
4. Crie o *script* SQL correspondente à alteração proposta com os comandos de criação e/ou alteração das tabelas normalizadas.

1. A tabela Cliente não segue a Primeira Forma Normal por quê ainda pode ser desmembrada em outras tabelas. Devemos fazer com que cada coluna contenha apenas um valor atômico, para isso devemos desmembrar valores não atômicos. Abaixo deixo um exemplo de como seria uma possível aplicação:

Id	Nome	Cpf	Endereco	Telefone
1	André	74084269069	Avenida Aníbal Pereira 01,Centro,Afligid os,BA	(96) 2881-0817, (95) 2915-5787
2	Marcos	82864770067	Rua Anitápolis 283,Centro,Alfre do Wagner,SC	(69) 2512-2802

Tabela Normalizada

Tabela de Alunos

Id	Nome	Cpf	Logradouro	Bairro	Cidade	Estado
1	André	74084269069	Avenida Aníbal Pereira 01	Centro	Afligidos	BA
2	Marco s	82864770067	Rua Anitápolis 283	Centro	Alfredo Wagner	SC

Tabela de Telefones

IdTelefone	Telefone	IdAluno
1	(96) 2881-0817	1
2	(95) 2915-5787	1
3	(69) 2512-2802	2

2. Na tabela Item_venda para aplicar a Segunda Forma Normal (2FN), precisamos garantir que a tabela esteja em conformidade com a Primeira Forma Normal (1FN) e a tabela Item_venda não atende, e os atributos não-chave não são totalmente dependente da chave primária. Para normalizar a tabela para a 2FN, precisamos dividir a tabela em duas ou mais tabelas, de modo que todos os atributos não-chave dependam totalmente da chave primária.

Abaixo deixo um exemplo de como seria uma possível aplicação:

Quantidade INT	valor_unitario	Subtotal	Nome_produto	venda_id	produto_id
1	10,33	10,33	bola	3	1
2	22,11	44,22	chinelo	4	2

Tabela Normalizada

1.Tabela de Venda (contendo informações sobre a venda e o produto):

venda_id	produto_id	Quantidade	valor_unitario	Subtotal
3	1	1	10,33	10,33
4	2	2	22,11	44,22

2.Tabela de Produto (contendo informações sobre os produtos):

produto_id	Nome_produto
1	bola
2	chinelo

3. Se data_pagamento depende de status (por exemplo, um pagamento só pode ser registrado se o status for "aprovado"), isso também indicaria uma violação da 3FN, pois status não é uma chave primária.

Tabela venda:

id	Data	data_envio	Status	valor_total	tipo_pagamento	numero_cartao_pagamento	data_pagamento	cliente_id
1								
2								

Como forma de normalização a estrutura da tabela “venda” poderia ser dividida em:

Tabela venda

id	data	data_envio	numero_cart ao_credito	valor_total	tipo_pagamento	cliente_id

Com tipo_pagamento (chave estrangeira para uma nova tabela de pagamento), cliente_id (chave estrangeira para uma tabela de clientes).

Tabela pagamento :

tipo_pagamento	status

4. O script de criação das tabelas normalizadas está colocado abaixo:

-- Tabela de Alunos

```
CREATE TABLE Alunos (  
  Id INT PRIMARY KEY,  
  Nome VARCHAR(100) NOT NULL,  
  Cpf VARCHAR(11) NOT NULL UNIQUE,  
  Logradouro VARCHAR(255) NOT NULL,  
  Bairro VARCHAR(100) NOT NULL,  
  Cidade VARCHAR(100) NOT NULL,  
  Estado VARCHAR(2) NOT NULL  
);
```

-- Tabela de Telefones

```
CREATE TABLE Telefones (  
  IdTelefone INT PRIMARY KEY,  
  Telefone VARCHAR(15) NOT NULL,  
  IdAluno INT,  
  FOREIGN KEY (IdAluno) REFERENCES Alunos(Id)  
);
```

-- Tabela de Produtos

```
CREATE TABLE Produtos (  
  produto_id INT PRIMARY KEY,  
  Nome_produto VARCHAR(100) NOT NULL  
);
```

-- Tabela de Vendas (contendo informações sobre a venda e o produto)

```
CREATE TABLE Vendas (  
    venda_id INT PRIMARY KEY,  
    produto_id INT,  
    Quantidade INT NOT NULL,  
    valor_unitario DECIMAL(10, 2) NOT NULL,  
    Subtotal DECIMAL(10, 2) NOT NULL,  
    FOREIGN KEY (produto_id) REFERENCES Produtos(produto_id)  
);
```

-- Tabela de Venda (informações sobre a venda)

```
CREATE TABLE Venda (  
    id INT PRIMARY KEY,  
    data DATETIME NOT NULL,  
    data_envio DATETIME,  
    numero_cartao_credito VARCHAR(16),  
    valor_total DECIMAL(10, 2) NOT NULL,  
    tipo_pagamento VARCHAR(50),  
    cliente_id INT,  
    FOREIGN KEY (cliente_id) REFERENCES Alunos(Id)  
);
```

-- Tabela de Pagamento

```
CREATE TABLE Pagamento (  
    tipo_pagamento VARCHAR(50) PRIMARY KEY,  
    status VARCHAR(50) NOT NULL  
);
```