

Exercícios – Segunda Forma Normal (2FN)

Exercício 1 – Conceito

Explique com suas próprias palavras o que significa uma tabela estar na Segunda Forma Normal.

Resposta: Para estar na Segunda Forma Normal(2FN) ela exige que a tabela esteja na Primeira Forma Normal(1FN), e todos os seus atributos não-chave devem ser totalmente dependentes da chave primária inteira, e não apenas uma parte dela.

Exercício 2 – Identificação

Considere a tabela abaixo:

PedidoID	ProdutoID	NomeProduto	Quantidade
001	101	Caneta	10
001	102	Lápis	5
002	103	Caderno	2

A tabela está em 2FN? Justifique sua resposta.

Resposta: Não, a tabela não está em 2FN, porque o **NomeProduto** e **Quantidade** estão dependendo apenas do **ProdutoID**. O motivo é que ela contém uma dependência funcional parcial, violando a regra que exige que todos os atributos não-chave dependam da chave primária completa.

Exercício 3 – Dependência Parcial

Explique o que é dependência parcial de uma coluna em relação à chave primária composta e por que isso viola a 2FN. Dê um exemplo.

Resposta: A dependência parcial ocorre quando um atributo (coluna) em uma tabela depende apenas de uma parte de uma chave primária composta, e não de uma chave primária inteira. E ela a viola pois a 2FN exige que todos os atributos não-chave de uma tabela sejam funcionalmente dependentes da chave primária completa.

Exercício 4 – Conversão para 2FN

Transforme a tabela do exercício 2 em tabelas que respeitem a 2FN, mantendo as informações corretamente.

Resposta:

Tabela 1 (ItensDoPedido)

Esta tabela armazena a relação entre pedidos, produtos e a quantidade.

PedidoID	ProdutoID	Quantidade
001	101	10
001	102	5
002	103	2

Resposta:

Tabela 2 (Produtos)

Esta tabela armazena as informações dos produtos, onde NomeProduto depende totalmente de ProdutoID.

ProdutoID	NomeProduto
101	Caneta
102	Lápis
103	Caderno

Exercício 5 – Tabela com Chave Composta

A tabela a seguir registra cursos e professores:

Cursol D	Professorl D	NomeProfesso r	Sal a
C01	P01	Ana	101
C01	P02	Bruno	101
C02	P03	Carla	102

Identifique as dependências parciais e refatore a tabela para 2FN.

Resposta:

Tabela: (Professor)

ProfessorID	NomeProfessor
P01	Ana
P02	Bruno
P03	Carla

Tabela: (Curso)

CursolD	Sala
C01	101
C02	102

Tabela: (Rel_Curso_Professor)

CursolD	ProfessorID
C01	P01

C01

P02

C02

P03

Exercício 6 – Benefícios da 2FN

Liste três benefícios de se normalizar uma tabela para a 2FN.

Resposta:

- 1. Redução de redundância:** Evita a repetição de dados ao eliminar dependências parciais.
- 2. Melhora a integridade dos dados:** Diminui o risco de Anomalias de atualização, inserção e deleção, pois as informações são armazenadas com um local único.
- 3. Aumenta o desempenho:** Organiza a base de dados de forma mais eficiente, facilitando buscas e manutenção.

Exercício 7 – Multivalorados e 2FN

Explique como a existência de campos multivalorados pode afetar a segunda forma normal e o que deve ser feito para corrigir.

Resposta: A existência de campos multivalorados(campos que contêm múltiplos valores), viola a Primeira Forma Normal (1FN), e consequentemente também impede que a tabela esteja na Segunda Forma Normal (2FN).

Para corrigir a situação e permitir a normalização para a 2FN e além, o campo multivalorado deve ser extraído para uma nova tabela.

- 1.Crie uma nova tabela:** Essa nova tabela armazenará os valores do atributo multivalorado. Ela terá uma chave primária própria e uma chave estrangeira que referencia a tabela original.
- 2.Mantenha a integridade:** A chave estrangeira fará o vínculo entre as duas tabelas.

Exercício 8 – Exercício Prático

Crie uma tabela de vendas de produtos que esteja em 2FN, considerando que a tabela terá chave primária composta (PedidoID + ProdutoID).

Resposta:

1. Tabela Produtos

Esta tabela armazena os detalhes dos produtos.

Chave Primária: ProdutoID

Campos: ProdutoID, NomeProduto, Preço

ProdutoID	NomeProduto	Preço
101	Caneta	1.50
102	Lápis	0.80
103	Caderno	5.00

2. Tabela Pedidos

Esta tabela armazena os detalhes dos pedidos.

Chave Primária: PedidoID

Campos: PedidoID, DataPedido, ClienteID

PedidoID	DataPedido	ClienteID
001	2025-10-27	500
002	2025-10-27	501

3. Tabela `ItensDoPedido`

Esta é a tabela de junção que representa a relação de muitos-para-muitos entre pedidos e produtos.

Chave Primária: (`PedidoID`, `ProdutoID`) (chave composta)

Campos: `PedidoID`, `ProdutoID`, `Quantidade`

PedidoID	ProdutoID	Quantidade
001	101	10
001	102	5
002	103	2

Exercício 9 – Teoria e Prática

Dê um exemplo de tabela que esteja na 1FN mas não esteja na 2FN, explicando a razão.

Resposta:

Tabela em 1FN, mas não em 2FN:

AlunoID	DisciplinaID	NomeAluno	NomeDisciplina	Nota
101	B12	João	Banco de Dados	8.5
101	C21	João	Redes	9.0

A chave primária é composta por (AlunoID, DisciplinaID). No entanto, NomeAluno depende apenas de AlunoID e NomeDisciplina depende apenas de DisciplinaID. Isso viola a 2FN porque os atributos não-chave (NomeAluno, NomeDisciplina) não dependem da chave primária completa, mas apenas de parte dela. Isso gera redundância e anomalias de atualização.

Exercício 10 – Revisão

Você recebeu a tabela abaixo:

VendaID	ProdutoID	NomeProduto	PrecoUnitario	Quantidade
V001	201	Caneta	2.50	10
V001	202	Lápis	1.50	5

1. Verifique se a tabela está em 2FN.
2. Caso não esteja, reescreva-a em 2FN, separando as informações corretamente.

Resposta:

A tabela não está em 2FN:

Pois a chave primária da tabela é composta por (VendaID, ProdutoID). No entanto, os atributos NomeProduto e PrecoUnitario não dependem da chave primária completa, mas apenas de parte dela (ProdutoID). Isso configura uma dependência funcional parcial, o que viola a regra da Segunda Forma Normal.

Reescrevendo a tabela em 2FN:

Para normalizar, a tabela original deve ser dividida em duas: uma para os dados dos produtos e outra para os itens das vendas, garantindo que cada atributo não-chave dependa da chave primária completa de sua respectiva tabela.

1. Tabela ItensDaVenda

Esta tabela armazena a relação entre as vendas, os produtos e a quantidade vendida.

Chave Primária: (VendaID, ProdutoID)

Campos: VendaID, ProdutoID, Quantidade

VendaID	ProdutoID	Quantidade
V001	201	10
V001	202	5

2. Tabela **Produtos**

Esta tabela armazena os dados dos produtos.

Chave Primária: ProdutoID

Campos: ProdutoID, NomeProduto, PrecoUnitario

ProdutoID	NomeProduto	PrecoUnitario
201	Caneta	2.50
202	Lápis	1.50