Exercício 1: Sistema de Vendas

Considere um banco de dados de um sistema de vendas com as seguintes tabelas:

Tabela Clientes:

| ClienteID | Nome | Endereço | Cidade |
| --- | --- | --- | --- |
| 101 | Maria Silva | Rua A, 123 | São Paulo |
| 102 | João Santos | Av. A, 145 | Curitiba |

Tabela Pedidos:

| PedidoID | ClienteID | Data | Total |
| --- | --- | --- | --- |
| 1001 | 101 | 2023-01-10 | R$ 150 |
| 1002 | 202 | 2023-01-15 | R$ 200 |

Tabela ItensPedido:

| PedidoID | Produto | Quantidade |
| --- | --- | --- |
| 1001 | Camiseta | 2 |
| 1001 | Calça | 1 |
| 1002 | Tênis | 1 |

Para melhorar o desempenho das consultas de vendas por cliente, você decide aplicar denormalização seletiva. Como você poderia reestruturar essas tabelas para otimizar as consultas?

Exercício 2: Catálogo de Produtos

Considere um sistema de comércio eletrônico com as seguintes tabelas:

Tabela Produtos:

| ProdutoID | Nome | Preço | Estoque |
| --- | --- | --- | --- |
| 101 | Camiseta | R$ 30 | 50 |
| 102 | Calça | RS 40 | 30 |

Tabela Categorias:

| CategoriaID | Nome |
| --- | --- |
| 1 | Roupas |
| 2 | Calçados |

Para melhorar a velocidade de pesquisa de produtos por categoria, você decide aplicar denormalização seletiva. Como você poderia reestruturar essas tabelas para otimizar as consultas?

Exercício 3: Sistema de Biblioteca

Considere um banco de dados de uma biblioteca com as seguintes tabelas:

Tabela Livros:

| ISBN | Titulo | Autor |
| --- | --- | --- |
| 978-1234567 | Introdução à História | João da Silva |
| 978-2345678 | Aprenda Python | Maria Santos |

Tabela Empréstimos:

| EmpréstimoID | ISBN | Data |
| --- | --- | --- |
| 1001 | 978-1234567 | 2023-01-10 |
| 1002 | 978-2345678 | 2023-01-15 |

Para melhorar o desempenho das consultas de empréstimos por livro, você decide aplicar denormalização seletiva. Como você poderia reestruturar essas tabelas para otimizar as consultas?