

Alunos: Fábio Emanuell Pereira Milagres, João Pedro de Almeida Dupim e Pedro Henrique Santos.

Exemplo 1:

(a.1) Enunciado ECA (Evento-Condição-Ações)

Evento: O sistema precisa visualizar um resumo das notas fiscais de venda, incluindo informações do cliente.

Condição: A consulta ocorre quando o usuário deseja visualizar os detalhes das notas fiscais de venda com os dados dos clientes associados.

Ações:

- Criar uma view que consolida informações da tabela **NF\_Venda** e da tabela **Cliente**.
- Exibir código da nota fiscal, data de emissão, valor total e informações do cliente (CPF, nome, cidade, bairro e rua).

(a.3) Documentação e Decisões de Projeto

1. Uso de View: A view vw\_nf\_venda\_resumo simplifica consultas frequentes, permitindo que os dados da nota fiscal de venda e do cliente sejam acessados de forma integrada.
2. Associação entre Tabelas: Utilizamos um JOIN entre NF\_Venda e Cliente para garantir que cada nota fiscal esteja vinculada ao seu respectivo cliente.
3. Campos Seleccionados: Optamos por exibir apenas os campos essenciais para um resumo, evitando redundâncias.
4. Compatibilidade: Criamos versões para MySQL e PostgreSQL, ajustando a sintaxe conforme as particularidades de cada banco.

(a.4) Casos de Teste Interessantes

Codigo_NF	Data_Emissao	Valor	CPF Cliente	Nome_Cliente	Cidade	Bairro	Rua
1001	2024-01-10	500.00	12345678901	João Silva	São Paulo	Centro	Av. Paulista
1002	2024-02-01	1500.00	98765432100	Maria Souza	Rio de Janeiro	Copacabana	Rua Atlântica
1003	2024-02-15	0.00	11122233344	Pedro Costa	Belo Horizonte	Savassi	Rua Sergipe

1. Caso 1: Nota fiscal registrada corretamente com um cliente de São Paulo.
2. Caso 2: Nota fiscal de valor elevado vinculada a um cliente do Rio de Janeiro.
3. Caso 3: Nota fiscal com valor zero para testar exibição de valores sem registros de compra.

## Exemplo 2:

### (a.1) Enunciado ECA (Evento-Condição-Ações)

Evento: O sistema precisa visualizar o estoque atual dos produtos considerando compras e vendas.

Condição: A consulta ocorre quando o usuário deseja verificar a quantidade de estoque disponível de cada produto.

Ações:

- Criar uma view que exiba o estoque inicial, total comprado, total vendido e estoque disponível.
- Utilizar sub consultas com funções de agregação para calcular os totais de compra e venda.

### (a.3) Documentação e Decisões de Projeto

1. Uso de View: A view vw\_produto\_estoque facilita a consulta do estoque atual de cada produto sem a necessidade de cálculos manuais.
2. Tratamento de Valores Nulos: O uso de IFNULL (MySQL) e COALESCE (PostgreSQL) garante que valores nulos sejam substituídos por zero para evitar erros de cálculo.
3. Subconsultas: Foram usadas subconsultas para calcular os totais comprados e vendidos de cada produto.
4. Compatibilidade: Foram feitas adaptações para PostgreSQL, incluindo aspas duplas para identificar nomes de colunas e tabelas.

### (a.4) Casos de Teste Interessantes

Codigo	Nome	Estoque_Inicial	Total_Comprado	Total_Vendido	Estoque_Disponivel
101	Caneta Azul	100	50	20	130
102	Caderno A4	200	30	50	180
103	Borracha	50	0	10	40

1. Caso 1: Produto com estoque inicial 100, recebeu 50 unidades compradas e vendeu 20, resultando em estoque de 130.
2. Caso 2: Produto com estoque inicial 200, recebeu 30 unidades e vendeu 50, resultando em estoque de 180.

3. Caso 3: Produto com estoque inicial 50, sem compras registradas e 10 vendas, resultando em estoque de 40.