Projeto de Banco de Dados: Sistema de Vendas

1. Introdução

Este projeto modela um banco de dados para um sistema de vendas e controle de estoque. Ele permite o cadastro de clientes e fornecedores, o gerenciamento de estoque e pedidos, além de registrar o histórico de vendas. A estrutura foi desenvolvida com o uso de objetos de banco (views, procedures, functions e triggers) para garantir a integridade e automação dos processos.

2. Estrutura do Banco

Banco: vendas12
Tabelas:
- Clientes
- Fornecedores
- Estoque
- Pedidos
- Historico_Pedidos

3. Script SQL Completo (Comentado)

```
-- Criação do banco e uso
CREATE DATABASE vendas12;
USE vendas12;

-- Tabela de clientes
CREATE TABLE Clientes (
    ID INT PRIMARY KEY AUTO_INCREMENT,
    Nome VARCHAR(100),
    Email VARCHAR(100)
);

-- Tabela de fornecedores
CREATE TABLE Fornecedores (
    ID INT PRIMARY KEY AUTO_INCREMENT,
    Nome VARCHAR(100),
    Contato VARCHAR(100)
);

-- Estoque de produtos
CREATE TABLE Estoque (
    ID INT PRIMARY KEY AUTO_INCREMENT,
    Produto VARCHAR(100),
    Quantidade INT,
    ID_Fornecedor INT,
    FOREIGN KEY (ID_Fornecedor) REFERENCES Fornecedores(ID)
);

-- Pedidos atuais
CREATE TABLE Pedidos (
    ID INT PRIMARY KEY AUTO_INCREMENT,
    Produto VARCHAR(100),
```

```
    Quantidade INT,
    ID_Cliente INT,
    Statuss VARCHAR(50) DEFAULT 'Pendente',
    DataPedido TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (ID_Cliente) REFERENCES Clientes(ID)
);

-- Histórico de pedidos confirmados
CREATE TABLE Historico_Pedidos (
    ID INT PRIMARY KEY AUTO_INCREMENT,
    Produto VARCHAR(100),
    Quantidade INT,
    ID_Cliente INT,
    Statuss VARCHAR(50),
    DataPedido TIMESTAMP,
    FOREIGN KEY (ID_Cliente) REFERENCES Clientes(ID)
);
```

4. Objetos do Banco

VIEW - Pedidos Confirmados

```
CREATE VIEW View_Pedidos_Confirmados AS
SELECT P.ID, C.Nome AS Cliente, P.Produto, P.Quantidade, P.Statuss, P.DataPedido
FROM Pedidos P
JOIN Clientes C ON P.ID_Cliente = C.ID
WHERE P.Statuss = 'Confirmado';
```

FUNCTION - Total de Pedidos por Cliente

```
CREATE FUNCTION TotalPedidosCliente(clienteID INT)
RETURNS INT
BEGIN
    DECLARE total INT;
    SELECT COUNT(*) INTO total FROM Pedidos WHERE ID_Cliente = clienteID;
    RETURN total;
END;
```

PROCEDURE - Fazer Pedido

```
CREATE PROCEDURE FazerPedido(IN nomeProduto VARCHAR(100), IN qtd INT, IN clienteID INT)
BEGIN
    DECLARE estoqueDisponivel INT;
    SELECT Quantidade INTO estoqueDisponivel FROM Estoque WHERE Produto = nomeProduto;
    IF estoqueDisponivel IS NOT NULL AND estoqueDisponivel >= qtd THEN
        INSERT INTO Pedidos (Produto, Quantidade, ID_Cliente)
        VALUES (nomeProduto, qtd, clienteID);
    ELSE
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Produto inexistente ou estoque insuficiente.';
    END IF;
END;
```

TRIGGER - Atualizar Estoque e Histórico

```
CREATE TRIGGER AtualizaEstoqueEHistorico
```

Projeto de Banco de Dados: Sistema de Vendas

```
AFTER UPDATE ON Pedidos
FOR EACH ROW
BEGIN
   IF NEW.Statuss = 'Confirmado' AND OLD.Statuss != 'Confirmado' THEN
      UPDATE Estoque SET Quantidade = Quantidade - NEW.Quantidade
      WHERE Produto = NEW.Produto;
      INSERT INTO Historico_Pedidos (Produto, Quantidade, ID_Cliente, Statuss, DataPedido)
      VALUES (NEW.Produto, NEW.Quantidade, NEW.ID_Cliente, NEW.Statuss, NEW.DataPedido);
   END IF;
END;
```

5. Dados de Teste (INSERTs)

```
-- Exemplo de inserções já incluídas no script original
```

6. Explicações Técnicas

- View simplifica visualização dos pedidos confirmados.
- Function retorna total de pedidos de um cliente.
- Procedure automatiza criação de pedidos com verificação de estoque.
- Trigger atualiza estoque e insere no histórico automaticamente após confirmação do pedido.

7. Conclusão

O sistema apresentado oferece um controle completo de pedidos e estoque, utilizando boas práticas de modelagem e automação em banco de dados. A integração de objetos SQL fortalece a consistência dos dados, tornando o sistema ideal para pequenas e médias empresas de varejo ou e-commerce que desejam uma gestão eficiente de vendas.