Relatório - Aula Prática: Programação e Desenvolvimento de Banco de Dados

# Introdução

A presente aula prática tem como objetivo desenvolver a estrutura de um banco de dados relacional utilizando o MySQL Workbench. As atividades incluíram a criação do banco de dados 'Loja', implementação das tabelas conforme modelo Entidade-Relacionamento (EER), inserção de dados e criação de uma View para consulta de contas não pagas.

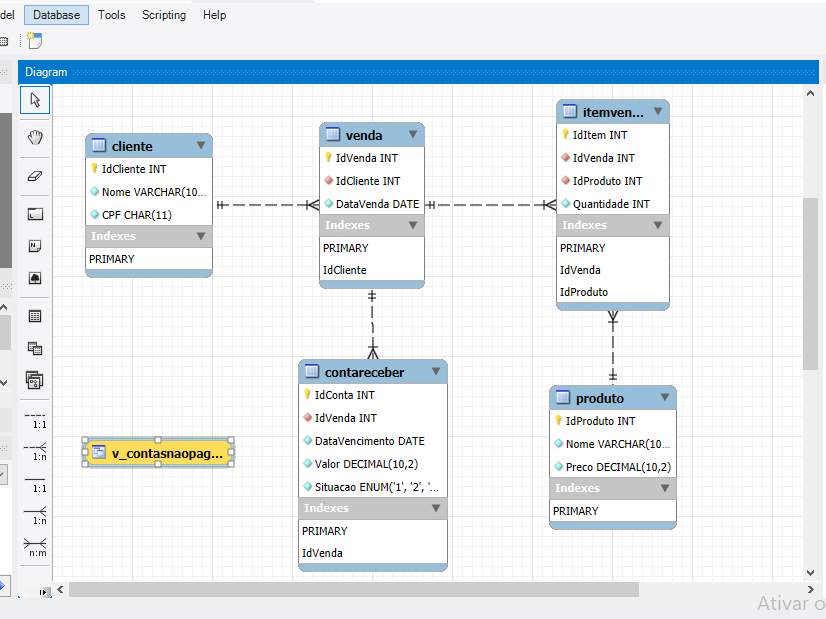
# Metodologia

Foram seguidas três etapas principais:  
1. Criação das tabelas conforme modelo físico extraído do DER.  
2. Inserção de dados de exemplo nas tabelas utilizando comandos DML.  
3. Criação de uma View para listar contas não pagas, com base no campo 'Situação' igual a 1.  
As ferramentas utilizadas foram o MySQL Community Server e o MySQL Workbench.

# Resultados

As figuras a seguir demonstram os resultados obtidos durante a execução das atividades:

Modelo Físico do Banco de Dados (Diagrama EER)



Criação das Tabelas no MySQL Workbench

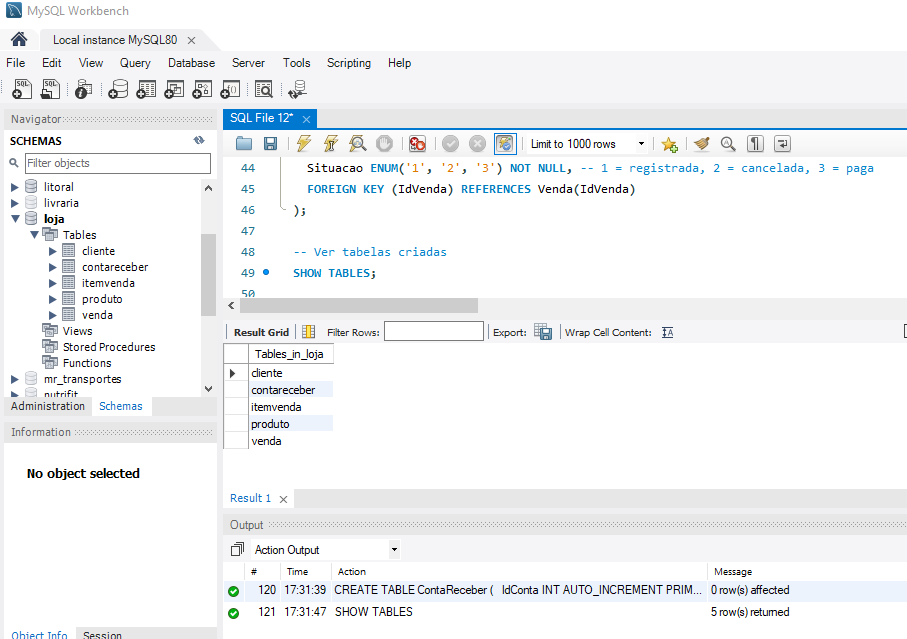


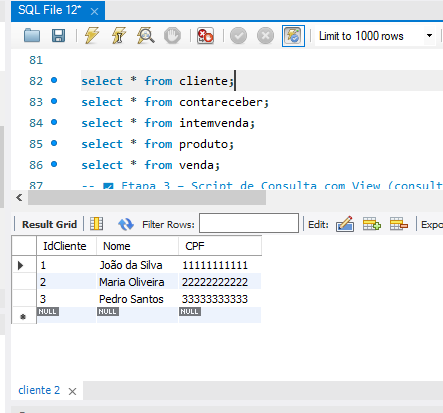
Tabela Cliente com registros inseridos

Tabela ContaReceber com registro inseridos

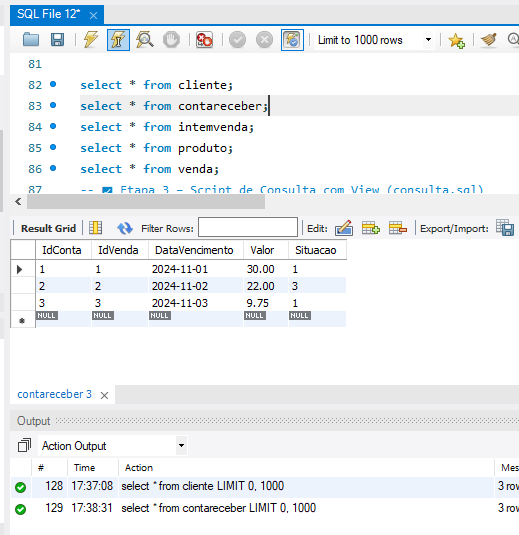


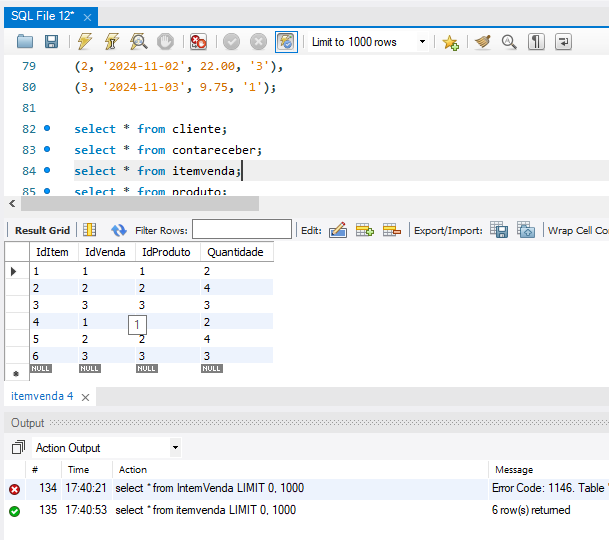
Tabela ItemVenda com registoros inseridos 

Tabela Produto com registros inseridos

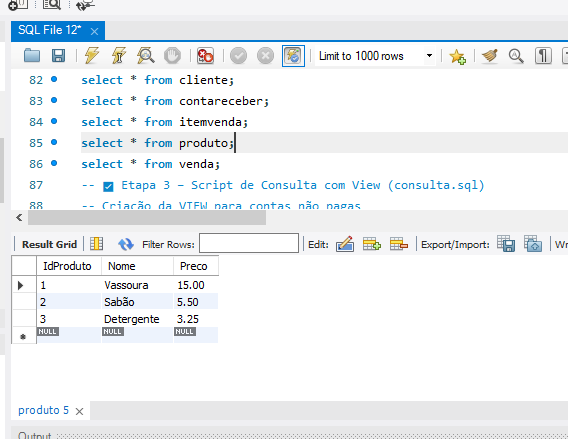
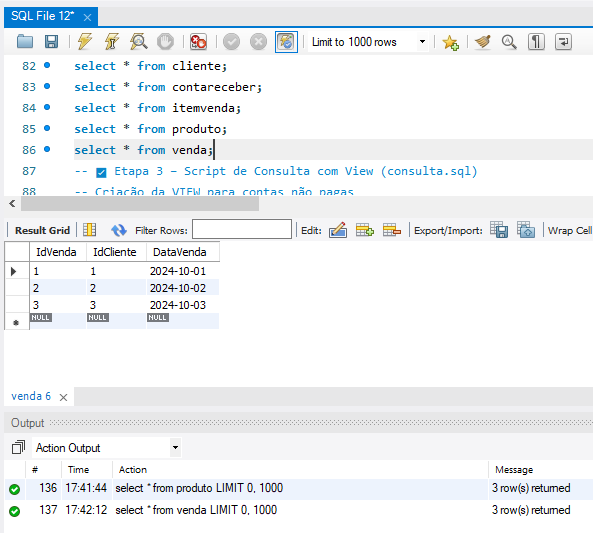
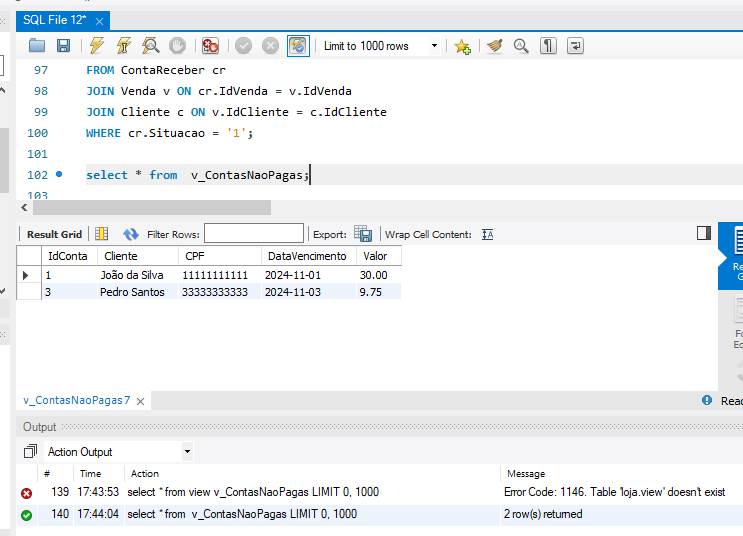


Tabela Venda com registros inseridos



View com contas não pagas



# Scripts Utilizados

Os scripts SQL utilizados para criar, inserir e consultar dados estão listados abaixo:

-- ✅ Etapa 1 – Script de Criação do Banco de Dados (loja.sql)  
-- Criação do banco de dados  
CREATE DATABASE Loja;  
USE Loja;  
  
-- Tabela Cliente  
CREATE TABLE Cliente (  
 IdCliente INT AUTO\_INCREMENT PRIMARY KEY,  
 Nome VARCHAR(100) NOT NULL,  
 CPF CHAR(11) NOT NULL  
);  
  
-- Tabela Produto  
CREATE TABLE Produto (  
 IdProduto INT AUTO\_INCREMENT PRIMARY KEY,  
 Nome VARCHAR(100) NOT NULL,  
 Preco DECIMAL(10,2) NOT NULL  
);

-- Tabela Venda  
CREATE TABLE Venda (  
 IdVenda INT AUTO\_INCREMENT PRIMARY KEY,  
 IdCliente INT NOT NULL,  
 DataVenda DATE NOT NULL,  
 FOREIGN KEY (IdCliente) REFERENCES Cliente(IdCliente)  
);  
  
-- Tabela ItemVenda  
CREATE TABLE ItemVenda (  
 IdItem INT AUTO\_INCREMENT PRIMARY KEY,  
 IdVenda INT NOT NULL,  
 IdProduto INT NOT NULL,  
 Quantidade INT NOT NULL,  
 FOREIGN KEY (IdVenda) REFERENCES Venda(IdVenda),  
 FOREIGN KEY (IdProduto) REFERENCES Produto(IdProduto)  
);  
  
-- Tabela ContaReceber  
CREATE TABLE ContaReceber (  
 IdConta INT AUTO\_INCREMENT PRIMARY KEY,  
 IdVenda INT NOT NULL,  
 DataVencimento DATE NOT NULL,  
 Valor DECIMAL(10,2) NOT NULL,  
 Situacao ENUM('1', '2', '3') NOT NULL, -- 1 = registrada, 2 = cancelada, 3 = paga  
 FOREIGN KEY (IdVenda) REFERENCES Venda(IdVenda)  
);  
  
-- Ver tabelas criadas  
SHOW TABLES;  
  
-- ✅ Etapa 2 – Script de Inserção de Dados (inserir.sql)  
-- Inserção de Clientes  
INSERT INTO Cliente (Nome, CPF) VALUES  
('João da Silva', '11111111111'),  
('Maria Oliveira', '22222222222'),  
('Pedro Santos', '33333333333');  
  
-- Inserção de Produtos  
INSERT INTO Produto (Nome, Preco) VALUES  
('Vassoura', 15.00),  
('Sabão', 5.50),  
('Detergente', 3.25);

-- Inserção de Vendas  
INSERT INTO Venda (IdCliente, DataVenda) VALUES  
(1, '2024-10-01'),  
(2, '2024-10-02'),  
(3, '2024-10-03');  
  
-- Inserção de Itens de Venda  
INSERT INTO ItemVenda (IdVenda, IdProduto, Quantidade) VALUES  
(1, 1, 2),  
(2, 2, 4),  
(3, 3, 3);  
  
-- Inserção de Contas a Receber  
INSERT INTO ContaReceber (IdVenda, DataVencimento, Valor, Situacao) VALUES  
(1, '2024-11-01', 30.00, '1'),  
(2, '2024-11-02', 22.00, '3'),  
(3, '2024-11-03', 9.75, '1');  
  
select \* from cliente;  
select \* from contareceber;  
select \* from itemvenda;  
select \* from produto;  
select \* from venda;  
  
-- ✅ Etapa 3 – Script de Consulta com View (consulta.sql)  
-- Criação da VIEW para contas não pagas  
CREATE VIEW v\_ContasNaoPagas AS  
SELECT   
 cr.IdConta,  
 c.Nome AS Cliente,  
 c.CPF,  
 cr.DataVencimento,  
 cr.Valor  
FROM ContaReceber cr  
JOIN Venda v ON cr.IdVenda = v.IdVenda  
JOIN Cliente c ON v.IdCliente = c.IdCliente  
WHERE cr.Situacao = '1';  
  
select \* from v\_ContasNaoPagas;

# Conclusão

A atividade prática permitiu a aplicação dos conceitos fundamentais de criação e manipulação de bancos de dados relacionais. As etapas foram executadas com sucesso, resultando em uma base de dados funcional e estruturada, demonstrando o entendimento dos comandos DDL, DML e DQL.