**Grupo 10**

Banco de Dados

Criação das Tabelas e Índices

-- Ativar foreign keys

PRAGMA foreign\_keys = ON;

-- Limpar tabelas e views existentes (se houver)

DROP VIEW IF EXISTS DoacoesDetalhes;

DROP TABLE IF EXISTS Noticias;

DROP TABLE IF EXISTS Relatorio;

DROP TABLE IF EXISTS Doacao;

DROP TABLE IF EXISTS Campanha;

DROP TABLE IF EXISTS Usuario;

DROP TABLE IF EXISTS Ouvidoria;

-- ========================================

-- CRIANDO TABELAS

-- ========================================

-- Tabela Usuario (base - sem dependências)

CREATE TABLE Usuario (

cd\_cliente INTEGER PRIMARY KEY AUTOINCREMENT,

nome\_completo TEXT NOT NULL,

telefone TEXT,

cpf TEXT UNIQUE,

cep TEXT,

nome\_usuario TEXT UNIQUE NOT NULL,

senha TEXT NOT NULL,

email TEXT UNIQUE NOT NULL

);

-- Tabela Campanha (base - sem dependências)

CREATE TABLE Campanha (

cd\_campanha INTEGER PRIMARY KEY AUTOINCREMENT,

nome\_campanha TEXT NOT NULL,

meta\_arrecadacao REAL,

inicio DATE,

fim DATE

);

-- Tabela Doacao (depende de Usuario e Campanha)

CREATE TABLE Doacao (

cd\_doacao INTEGER PRIMARY KEY AUTOINCREMENT,

cd\_cliente INTEGER NOT NULL,

cd\_campanha INTEGER NOT NULL,

nome\_doacao TEXT NOT NULL,

tipo\_doacao TEXT NOT NULL,

forma\_arrecadacao TEXT,

status\_arrecadacao TEXT DEFAULT 'Pendente',

CONSTRAINT fk\_doacao\_usuario FOREIGN KEY (cd\_cliente)

REFERENCES Usuario(cd\_cliente),

CONSTRAINT fk\_doacao\_campanha FOREIGN KEY (cd\_campanha)

REFERENCES Campanha(cd\_campanha)

);

-- Tabela Relatorio (depende de Campanha)

CREATE TABLE Relatorio (

cd\_relatorio INTEGER PRIMARY KEY AUTOINCREMENT,

cd\_campanha INTEGER NOT NULL,

tipo\_relatorio TEXT NOT NULL,

valor\_gasto REAL,

data\_relatorio DATE NOT NULL,

caminho\_pdf TEXT, -- Nova coluna para o caminho do arquivo PDF

CONSTRAINT fk\_relatorio\_campanha FOREIGN KEY (cd\_campanha)

REFERENCES Campanha(cd\_campanha)

);

-- Tabela Noticias (depende de Campanha)

CREATE TABLE Noticias (

cd\_noticias INTEGER PRIMARY KEY AUTOINCREMENT,

cd\_campanha INTEGER NOT NULL,

titulo\_noticia TEXT NOT NULL,

data\_noticia DATE NOT NULL,

autor TEXT,

conteudo TEXT,

imagem TEXT, -- Nova coluna para o caminho da imagem

CONSTRAINT fk\_noticias\_campanha FOREIGN KEY (cd\_campanha)

REFERENCES Campanha(cd\_campanha)

);

-- Nova tabela para a ouvidoria

CREATE TABLE Ouvidoria (

cd\_mensagem INTEGER PRIMARY KEY AUTOINCREMENT,

nome\_remetente TEXT NOT NULL,

email\_remetente TEXT NOT NULL,

assunto TEXT,

mensagem TEXT NOT NULL,

data\_envio DATE NOT NULL

);

-- ========================================

-- CRIANDO ÍNDICES PARA PERFORMANCE

-- ========================================

-- Índices para Usuario

CREATE INDEX idx\_usuario\_email ON Usuario(email);

CREATE INDEX idx\_usuario\_telefone ON Usuario(telefone);

CREATE INDEX idx\_usuario\_cpf ON Usuario(cpf);

-- Índices para Doacao

CREATE INDEX idx\_doacao\_cliente ON Doacao(cd\_cliente);

CREATE INDEX idx\_doacao\_campanha ON Doacao(cd\_campanha);

CREATE INDEX idx\_doacao\_status ON Doacao(status\_arrecadacao);

-- Índices para Relatorio

CREATE INDEX idx\_relatorio\_campanha ON Relatorio(cd\_campanha);

-- Índices para Noticias

CREATE INDEX idx\_noticias\_campanha ON Noticias(cd\_campanha);

CREATE INDEX idx\_noticias\_data ON Noticias(data\_noticia);

-- Índices para Campanha

CREATE INDEX idx\_campanha\_nome ON Campanha(nome\_campanha);

-- ========================================

-- CRIANDO VIEW (Requisito da disciplina Banco de Dados)

-- ========================================

-- View para visualizar doações com informações do doador e da campanha

CREATE VIEW DoacoesDetalhes AS

SELECT

D.cd\_doacao,

D.nome\_doacao,

D.tipo\_doacao,

D.forma\_arrecadacao,

D.status\_arrecadacao,

U.nome\_completo AS nome\_doador,

U.email AS email\_doador,

C.nome\_campanha

FROM Doacao AS D

INNER JOIN Usuario AS U ON D.cd\_cliente = U.cd\_cliente

INNER JOIN Campanha AS C ON D.cd\_campanha = C.cd\_campanha;

-- ========================================

-- CRIANDO INSERT PARA TESTES

-- ========================================

-- Inserir dados de teste nas tabelas originais

INSERT INTO Usuario (nome\_completo, telefone, cpf, nome\_usuario, senha, email) VALUES

('João Silva Santos', '11999999999', '12345678901', 'joao123', 'senha123', 'joao@email.com'),

('Maria Oliveira', '11888888888', '98765432100', 'maria456', 'senha456', 'maria@email.com');

INSERT INTO Campanha (nome\_campanha, meta\_arrecadacao, inicio, fim) VALUES

('Campanha do Agasalho 2024', 50000.00, '2024-06-01', '2024-08-31'),

('Natal Solidário', 30000.00, '2024-11-01', '2024-12-25');

INSERT INTO Doacao (cd\_cliente, cd\_campanha, nome\_doacao, tipo\_doacao, forma\_arrecadacao) VALUES

(1, 1, 'Doação de Roupas', 'Roupas', 'Entrega'),

(2, 1, 'Doação em Dinheiro', 'Dinheiro', 'PIX');

-- Inserir dados de teste nas novas tabelas

INSERT INTO Ouvidoria (nome\_remetente, email\_remetente, assunto, mensagem, data\_envio) VALUES

('Pedro Martins', 'pedro@email.com', 'Dúvida sobre Doação', 'Olá, gostaria de saber como fazer uma doação com cartão de crédito.', '2025-09-18');

INSERT INTO Relatorio (cd\_campanha, tipo\_relatorio, valor\_gasto, data\_relatorio, caminho\_pdf) VALUES

(1, 'Relatório de Prestação de Contas', 45000.50, '2024-09-15', '/relatorios/prestacao-de-contas-2024.pdf');

INSERT INTO Noticias (cd\_campanha, titulo\_noticia, data\_noticia, autor, conteudo, imagem) VALUES

(1, 'Primeira Etapa da Campanha Concluída', '2024-07-20', 'Equipe Alma', 'Concluímos a primeira fase da campanha de agasalho com sucesso.', '/imagens/campanha-agasalho-fase1.jpg');

-- ========================================

-- VERIFICAÇÃO DAS TABELAS

-- ========================================

-- Verificar criação

SELECT 'Tabelas e índices criados com sucesso!' as resultado;

SELECT name FROM sqlite\_master WHERE type='table' ORDER BY name;

SELECT name FROM sqlite\_master WHERE type='view' ORDER BY name;

SELECT name, tbl\_name FROM sqlite\_master WHERE type='index' AND name LIKE 'idx\_%' ORDER BY tbl\_name;