**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS**

**CIÊNCIA DA COMPUTAÇÃO**

ALESSANDRA FARIA RODRIGUES

**TRABALHO PRÁTICO**

MODELAGEM DE BANCO DE DADOS

Belo Horizonte

2025

1. **Introdução e Contexto**

A solidariedade e o apoio às pessoas em situação de vulnerabilidade são ações essenciais em diversas comunidades. Nesse cenário, a instituição religiosa católica, **Comunidade São Sebastião - Paróquia Maria, Mãe dos Pobres**, mantém um programa social contínuo para auxiliar famílias em situação de vulnerabilidade na comunidade local. O principal objetivo do programa é a arrecadação, organização e distribuição de doações, como alimentos não perecíveis, roupas, cobertores e produtos de higiene.

Atualmente, o controle desse processo é feito de forma manual, o que gera dificuldades no gerenciamento do estoque, na identificação das necessidades dos assistidos e na prestação de contas. O objetivo deste projeto é modelar um banco de dados relacional para sistematizar e otimizar todo o ciclo de doações, garantindo um processo mais eficiente, transparente e justo.

1. **Descrição do Minimundo**

Esta é a descrição textual de minimundo do **Sistema de Doações**. O Sistema de Doações gerencia o processo de doações de objetos essenciais para pessoas em situação de vulnerabilidade social. Este sistema é projetado para a Comunidade São Sebastião - Paróquia Maria, Mãe dos Pobres, permitindo o gerenciamento de forma eficiente das doações, dos doadores e dos beneficiários, a fim de atender melhor às suas necessidades.

No centro da comunidade, a **Instituição** desenvolve um trabalho social vital, focado em arrecadar, organizar e distribuir doações para famílias em situação de vulnerabilidade. Os doadores podem deixar os objetos doados no próprio endereço da instituição, que serão posteriormente encaminhados aos beneficiários, a instituição também possui telefone para contato e CNPJ. A organização das doações é feita por **Voluntários**, membros da própria comunidade que são devidamente cadastrados com seus dados (nome, CPF, telefone) e são os responsáveis por registrar todas as movimentações de entrada e saída de doações.

No contexto desse sistema, o **Doador** é toda pessoa física ou jurídica que contribui com itens. Para cada doador, o sistema armazena informações de identificação e contato, como nome, CPF/CNPJ, e-mail e telefone, permitindo à instituição manter uma comunicação ativa e um registro claro da origem dos bens, porém, o doador também pode ser um doador anônimo. Um mesmo doador pode realizar múltiplas doações ao longo do tempo, estabelecendo uma relação de parceria contínua com a paróquia.

Cada **Doação** é tratada como um evento único, registrado com a data em que ocorreu e vinculada ao doador que a realizou. Neste momento, para um controle preciso do que foi recebido, o sistema detalha todos os **Itens** que compõem essa doação por meio do nome e de uma descrição. Um mesmo Item - por exemplo, "Arroz 1kg" - é um tipo de produto catalogado que pode ser recebido em diversas doações diferentes, e uma única doação pode, evidentemente, conter múltiplos itens. Por isso, o sistema registra a quantidade de cada item recebido naquela doação específica. Para itens perecíveis, a data de validade daquele lote também é uma informação crucial registrada no momento da entrada. Para facilitar a organização e o gerenciamento do estoque, todo Item é associado a uma **Categoria** pré-definida com o nome, como "Alimentos", "Higiene Pessoal" ou "Roupas".

Na outra parte do processo estão os **Receptores**, também chamados de beneficiários. São indivíduos e famílias da comunidade cujo estado de necessidade foi previamente avaliado e que estão cadastrados no sistema. Para cada receptor, são armazenadas informações essenciais como nome, CPF, endereço, telefone e composição familiar, permitindo que a distribuição seja direcionada e justa. Um mesmo receptor pode ser assistido diversas vezes, e o sistema mantém o histórico de toda a ajuda fornecida.

A entrega dessa ajuda é formalizada através de uma **Transação de Doação**. Este é o evento de saída, registrado com a data da entrega. Assim como na entrada, o sistema detalha quais itens e em qual quantidade foram entregues nesta transação específica, permitindo uma baixa automática e precisa do estoque e garantindo um registro completo e transparente de todo o ciclo da doação.

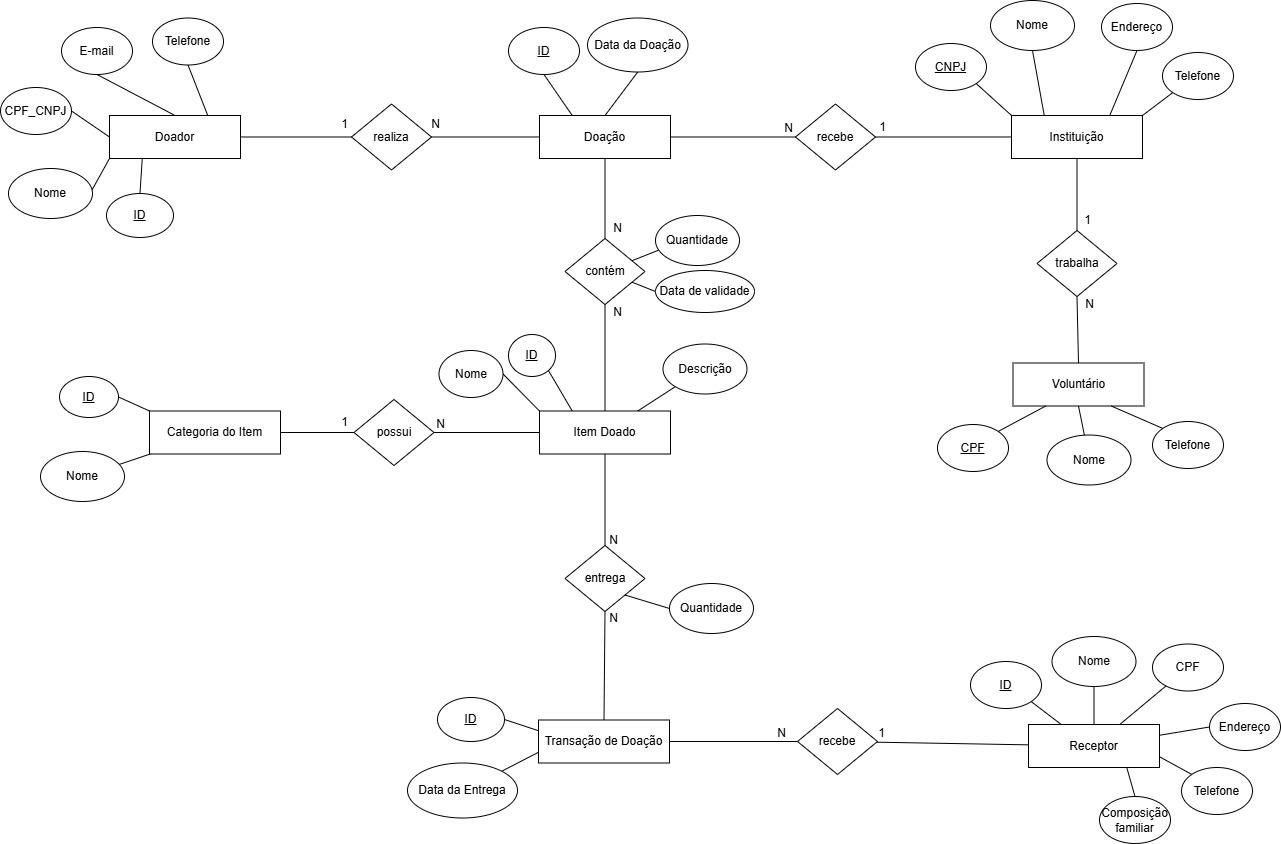
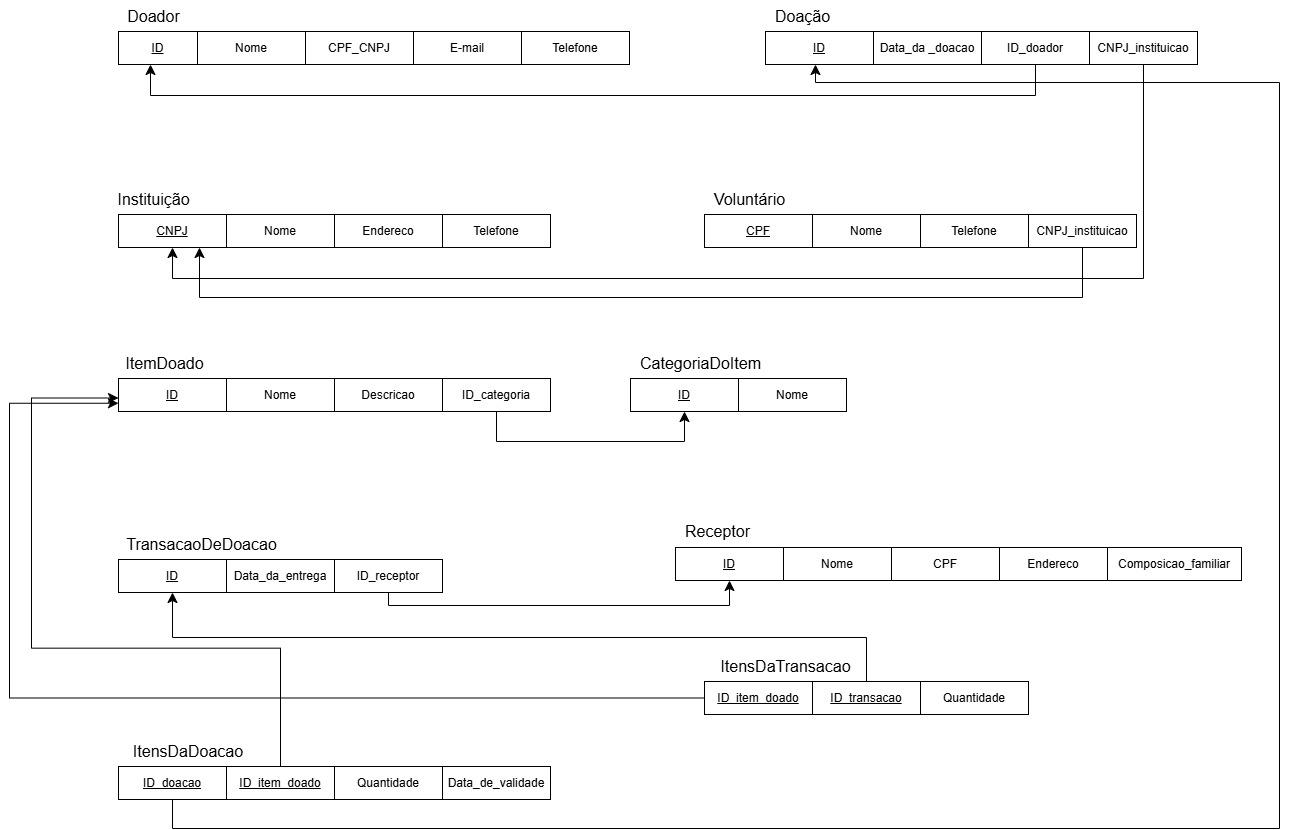
1. **Modelo Conceitual** 

Figura 1: Diagrama Entidade-Relacionamento do Sistema de Doações

1. **Modelo Lógico**

Figura 2: Diagrama de Esquema do Sistema de Doações

1. **Modelo de Implementação**

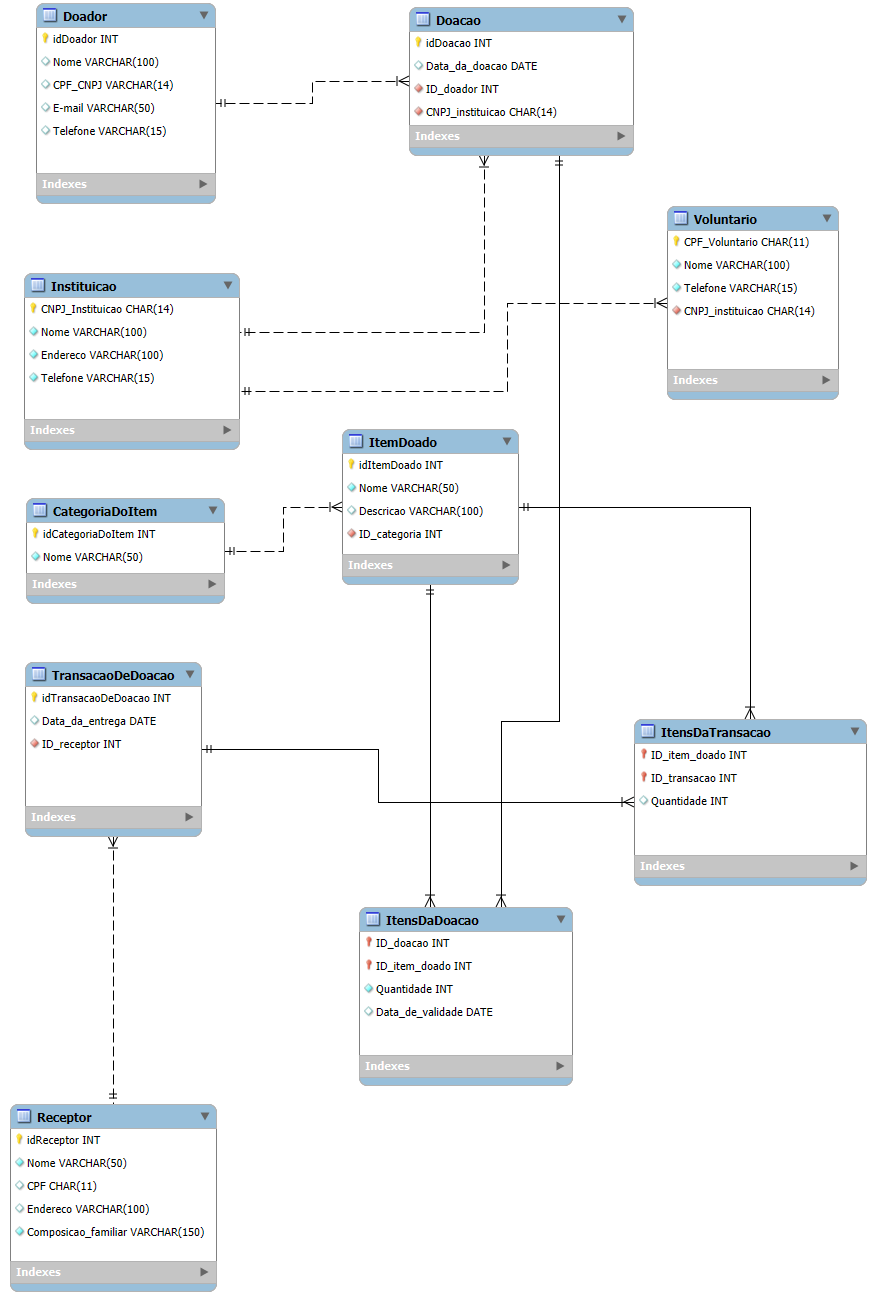
****

Figura 3: Diagrama de Implementação em Notação Pé-de-Galinha do Sistema de Doações

1. **Script SQL para criação do banco de dados**

-- MySQL Script generated by MySQL Workbench

-- Sat Jun 28 17:24:41 2025

-- Model: New Model Version: 1.0

-- MySQL Workbench Forward Engineering

SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;

SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION';

-- -----------------------------------------------------

-- Schema mydb

-- -----------------------------------------------------

CREATE DATABASE IF NOT EXISTS doacoes;

use doacoes;

-- -----------------------------------------------------

-- Table `CategoriaDoItem`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `CategoriaDoItem` ;

CREATE TABLE IF NOT EXISTS `CategoriaDoItem` (

`idCategoriaDoItem` INT NOT NULL,

`Nome` VARCHAR(50) NOT NULL,

PRIMARY KEY (`idCategoriaDoItem`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `Doador`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `Doador` ;

CREATE TABLE IF NOT EXISTS `Doador` (

`idDoador` INT NOT NULL,

`Nome` VARCHAR(100) NULL,

`CPF\_CNPJ` VARCHAR(14) NULL,

`E-mail` VARCHAR(50) NULL,

`Telefone` VARCHAR(15) NULL,

PRIMARY KEY (`idDoador`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `Instituicao`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `Instituicao` ;

CREATE TABLE IF NOT EXISTS `Instituicao` (

`CNPJ\_Instituicao` CHAR(14) NOT NULL,

`Nome` VARCHAR(100) NOT NULL,

`Endereco` VARCHAR(100) NOT NULL,

`Telefone` VARCHAR(15) NOT NULL,

PRIMARY KEY (`CNPJ\_Instituicao`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `Receptor`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `Receptor` ;

CREATE TABLE IF NOT EXISTS `Receptor` (

`idReceptor` INT NOT NULL,

`Nome` VARCHAR(50) NOT NULL,

`CPF` CHAR(11) NULL,

`Endereco` VARCHAR(100) NULL,

`Composicao\_familiar` VARCHAR(150) NOT NULL,

PRIMARY KEY (`idReceptor`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `Doacao`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `Doacao` ;

CREATE TABLE IF NOT EXISTS `Doacao` (

`idDoacao` INT NOT NULL,

`Data\_da\_doacao` DATE NULL,

`ID\_doador` INT NOT NULL,

`CNPJ\_instituicao` CHAR(14) NOT NULL,

PRIMARY KEY (`idDoacao`),

CONSTRAINT `fk\_Doacao\_Doador`

FOREIGN KEY (`ID\_doador`)

REFERENCES `Doador` (`idDoador`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Doacao\_Instituicao`

FOREIGN KEY (`CNPJ\_instituicao`)

REFERENCES `Instituicao` (`CNPJ\_Instituicao`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

CREATE INDEX `idx\_Doacao\_ID\_doador` ON `Doacao` (`ID\_doador`);

CREATE INDEX `idx\_Doacao\_CNPJ\_instituicao` ON `Doacao` (`CNPJ\_instituicao`);

-- -----------------------------------------------------

-- Table `ItemDoado`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `ItemDoado` ;

CREATE TABLE IF NOT EXISTS `ItemDoado` (

`idItemDoado` INT NOT NULL,

`Nome` VARCHAR(50) NOT NULL,

`Descricao` VARCHAR(100) NULL,

`ID\_categoria` INT NOT NULL,

PRIMARY KEY (`idItemDoado`),

CONSTRAINT `fk\_ItemDoado\_Categoria`

FOREIGN KEY (`ID\_categoria`)

REFERENCES `CategoriaDoItem` (`idCategoriaDoItem`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

CREATE INDEX `idx\_ItemDoado\_ID\_categoria` ON `ItemDoado` (`ID\_categoria`);

-- -----------------------------------------------------

-- Table `Voluntario`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `Voluntario` ;

CREATE TABLE IF NOT EXISTS `Voluntario` (

`CPF\_Voluntario` CHAR(11) NOT NULL,

`Nome` VARCHAR(100) NOT NULL,

`Telefone` VARCHAR(15) NOT NULL,

`CNPJ\_instituicao` CHAR(14) NOT NULL,

PRIMARY KEY (`CPF\_Voluntario`),

CONSTRAINT `fk\_Voluntario\_Instituicao`

FOREIGN KEY (`CNPJ\_instituicao`)

REFERENCES `Instituicao` (`CNPJ\_Instituicao`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

CREATE INDEX `idx\_Voluntario\_CNPJ\_instituicao` ON `Voluntario` (`CNPJ\_instituicao`);

-- -----------------------------------------------------

-- Table `TransacaoDeDoacao`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `TransacaoDeDoacao` ;

CREATE TABLE IF NOT EXISTS `TransacaoDeDoacao` (

`idTransacaoDeDoacao` INT NOT NULL,

`Data\_da\_entrega` DATE NULL,

`ID\_receptor` INT NOT NULL,

PRIMARY KEY (`idTransacaoDeDoacao`),

CONSTRAINT `fk\_Transacao\_Receptor`

FOREIGN KEY (`ID\_receptor`)

REFERENCES `Receptor` (`idReceptor`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

CREATE INDEX `idx\_TransacaoDeDoacao\_ID\_receptor` ON `TransacaoDeDoacao` (`ID\_receptor`);

-- -----------------------------------------------------

-- Table `ItensDaDoacao`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `ItensDaDoacao` ;

CREATE TABLE IF NOT EXISTS `ItensDaDoacao` (

`ID\_doacao` INT NOT NULL,

`ID\_item\_doado` INT NOT NULL,

`Quantidade` INT NOT NULL,

`Data\_de\_validade` DATE NULL,

PRIMARY KEY (`ID\_doacao`, `ID\_item\_doado`),

CONSTRAINT `fk\_ItensDaDoacao\_Doacao`

FOREIGN KEY (`ID\_doacao`)

REFERENCES `Doacao` (`idDoacao`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_ItensDaDoacao\_ItemDoado`

FOREIGN KEY (`ID\_item\_doado`)

REFERENCES `ItemDoado` (`idItemDoado`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

CREATE INDEX `idx\_ItensDaDoacao\_ID\_doacao` ON `ItensDaDoacao` (`ID\_doacao`);

CREATE INDEX `idx\_ItensDaDoacao\_ID\_item\_doado` ON `ItensDaDoacao` (`ID\_item\_doado`);

-- -----------------------------------------------------

-- Table `ItensDaTransacao`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `ItensDaTransacao` ;

CREATE TABLE IF NOT EXISTS `ItensDaTransacao` (

`ID\_item\_doado` INT NOT NULL,

`ID\_transacao` INT NOT NULL,

`Quantidade` INT NULL,

PRIMARY KEY (`ID\_item\_doado`, `ID\_transacao`),

CONSTRAINT `fk\_ItensDaTransacao\_ItemDoado`

FOREIGN KEY (`ID\_item\_doado`)

REFERENCES `ItemDoado` (`idItemDoado`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_ItensDaTransacao\_Transacao`

FOREIGN KEY (`ID\_transacao`)

REFERENCES `TransacaoDeDoacao` (`idTransacaoDeDoacao`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

CREATE INDEX `idx\_ItensDaTransacao\_ID\_item\_doado` ON `ItensDaTransacao` (`ID\_item\_doado`);

CREATE INDEX `idx\_ItensDaTransacao\_ID\_transacao` ON `ItensDaTransacao` (`ID\_transacao`);

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

1. **Script SQL para população do banco de dados**

--- script para populacao inicial e ficticia do banco de dados

use doacoes;

INSERT INTO CategoriaDoItem (idCategoriaDoItem, Nome) VALUES

(1, 'Alimentos'),

(2, 'Higiene'),

(3, 'Roupas');

INSERT INTO Doador (idDoador, Nome, CPF\_CNPJ, `E-mail`, Telefone) VALUES

(1, 'João Silva', '12345678901', 'joao@gmail.com', '31999990000'),

(2, 'Empresa Solidária', '12345678000199', 'contato@solidaria.com.br', '3133334444');

INSERT INTO Instituicao (CNPJ\_Instituicao, Nome, Endereco, Telefone) VALUES

('11111111000111', 'Paroquia', 'Rua das Flores, 100', '3122223344');

INSERT INTO Receptor (idReceptor, Nome, CPF, Endereco, Composicao\_familiar) VALUES

(1, 'Maria Souza', '98765432100', 'Rua da Paz, 45', 'Mãe solteira com 2 filhos'),

(2, 'Carlos Lima', '87654321099', 'Rua do Sol, 80', 'Casado, 3 filhos');

INSERT INTO ItemDoado (idItemDoado, Nome, Descricao, ID\_categoria) VALUES

(1, 'Arroz 5kg', 'Pacote de arroz branco tipo 1', 1),

(2, 'Sabonete', 'Sabonete neutro', 2),

(3, 'Camiseta M', 'Camiseta tamanho M', 3);

INSERT INTO Doacao (idDoacao, Data\_da\_doacao, ID\_doador, CNPJ\_instituicao) VALUES

(1, '2025-06-01', 1, '11111111000111'),

(2, '2025-06-10', 2, '11111111000111');

INSERT INTO Voluntario (CPF\_Voluntario, Nome, Telefone, CNPJ\_instituicao) VALUES

('99988877766', 'Ana Paula', '31988887777', '11111111000111'),

('88877766655', 'Ricardo Gomes', '31977776666', '11111111000111');

INSERT INTO TransacaoDeDoacao (idTransacaoDeDoacao, Data\_da\_entrega, ID\_receptor) VALUES

(1, '2025-06-15', 1),

(2, '2025-06-20', 2);

INSERT INTO ItensDaDoacao (ID\_doacao, ID\_item\_doado, Quantidade, Data\_de\_validade) VALUES

(1, 1, 2, '2026-01-01'),

(1, 2, 5, '2025-12-15'),

(2, 3, 3, NULL);

INSERT INTO ItensDaTransacao (ID\_item\_doado, ID\_transacao, Quantidade) VALUES

(1, 1, 1),

(2, 1, 2),

(3, 2, 2);