

TRABALHO PRÁTICO DE BANCO DE DADOS - RAQUEL DE PARDE MOTTA 2025/1

Esta é a descrição textual de minimundo do BANCAPE (Banco de Dados da Casa de Acolhida Padre Eustáquio). O BANCAPE gerencia o cadastro de crianças e seus responsáveis, as hospedagens, a recepção e distribuição de doações e os funcionários e voluntários que trabalham na ONG.

A CAPE (Casa de Acolhida Padre Eustáquio) é uma instituição não governamental que hospeda temporariamente crianças em situação de tratamento médico, como casos oncológicos, hematológicos, síndromes raras ou outras condições não infecciosas. A CAPE não oferece tratamento médico, funcionando como uma casa de apoio para pacientes vindos de outras cidades que realizam tratamento ambulatorial em Belo Horizonte.

Toda criança acolhida na CAPE (de 0 a 19 anos) obrigatoriamente deve estar acompanhada de um adulto responsável (maior de 18 anos), que permanecerá com ela durante a estadia. Cada criança é identificada por seu CPF e será cadastrada com nome, data de nascimento, endereço, categoria da doença (como oncologia, hematologia, entre outras) e status de acompanhamento (ativo, alta ou falecido). O responsável também é registrado com seu CPF como identificador, nome, telefone, e-mail e parentesco com a criança. Um adulto só pode ser responsável por uma única criança, enquanto cada criança tem exatamente um responsável.

A hospedagem na CAPE é organizada por quartos, que são identificados por um número único. Cada quarto é registrado com nome (a CAPE nomeia os quartos com nomes de pássaros), capacidade e ocupação de crianças atual. A capacidade diz respeito ao número de crianças que o quarto pode comportar - por exemplo, se um quarto possui 4 camas, a sua capacidade é de abrigar 2 crianças, pois cada uma delas é acompanhada por um adulto. A hospedagem de uma criança em um quarto é registrada por meio de uma estadia, identificada por um código único, que contém data de entrada e data de saída correspondentes à permanência no local. Cada estadia corresponde à hospedagem de uma única criança, alocada em um único quarto, mas crianças podem se hospedar várias vezes ao longo do tempo, acumulando estadias. Cada estadia corresponde a um único quarto, mas um quarto pode alocar a estadia de diversas crianças. O responsável é vinculado diretamente à criança e, portanto, não participa do cadastro de estadia.

A criança chega até a CAPE encaminhada por um hospital parceiro, que entra em contato com o setor social da CAPE para solicitar a hospedagem. Cada hospital parceiro é identificado por seu CNPJ e cadastrado com nome, cidade, e-mail e telefone de contato. Um hospital pode encaminhar várias crianças, mas cada criança é encaminhada por um único hospital.

A instituição é mantida exclusivamente por doações monetárias e de itens, não comercializando nenhum produto. Os doadores da CAPE podem ser pessoas físicas ou jurídicas. Um doador é identificado por um código único e cadastrado com seu nome, CPF ou CNPJ, endereço, telefone e e-mail. Doadores realizam doações. A doação é

registrada com os seguintes dados: código identificador, tipo (monetária ou item), data de recebimento pela CAPE e valor (quando de tipo monetário). A doação ter um doador associado a ela é opcional, doações podem ser registradas anonimamente caso sejam menores e esporádicas ou o doador não queira se identificar, sem necessidade de cadastro do doador. Um doador pode realizar várias doações, mas cada doação é feita por um único doador.

As doações que não são do tipo monetário contém itens de doação, que compõem o estoque da CAPE. Uma doação pode conter vários itens, mas cada item tem origem em uma única doação. Cada item é identificado por um código único. Além disso, cada item possui uma descrição, categoria (roupas, alimentos, brinquedos, produtos de higiene pessoal ou material de limpeza), quantidade, data de validade (opcional) e tamanho (opcional).

A distribuição dos itens arrecadados pela CAPE para os destinatários finais é registrada no sistema como uma entrega. A entrega, recebida pelo destinatário, possui um código identificador e uma data. É necessário registrar a quantidade de cada item que foi entregue na distribuição, pois um mesmo item pode participar de diversas entregas com quantidades diferentes, até que seu estoque acabe. Uma entrega pode distribuir vários itens. Uma entrega só pode ter um destino, mas um receptor pode receber várias entregas. Os destinatários podem ser as próprias crianças acolhidas pela instituição, a família da criança que permanece na cidade de origem (sendo que, nesses casos, a criança continua sendo registrada como receptora) ou ainda ONGs colaboradoras cadastradas. Essas ONGs são identificadas por seu CNPJ e cadastradas com nome, endereço, telefone e e-mail. Dessa forma, o sistema acompanha o caminho de cada item recebido e repassado pela CAPE, desde sua entrada no estoque até sua destinação final.

Os funcionários da CAPE são registrados com vínculo empregatício. Toda entrega possui um funcionário da CAPE como responsável, sendo que um mesmo funcionário pode se responsabilizar por várias entregas diferentes ao longo do tempo. Cada funcionário é identificado por seu CPF e cadastrado com nome, data de nascimento, telefone, e-mail, endereço, salário, tipo (CLT, prestador de serviço ou terceirizado), cargo, data de contratação e descrição dos benefícios que recebe (como plano de saúde, opcional). Funcionários podem ser responsáveis por doações e também supervisionar voluntários.

Voluntários não possuem vínculo empregatício e portanto são registrados de forma diferente dos funcionários. Eles são cadastrados com seu CPF (identificador), nome, endereço, telefone, e-mail, periodicidade (se aplicável), datas de colaboração e função desempenhada em determinada data. Todo voluntário possui um funcionário responsável por sua supervisão, sendo que um mesmo funcionário pode acompanhar vários voluntários.

DIAGRAMA ENTIDADE-RELACIONAMENTO BANCAPE

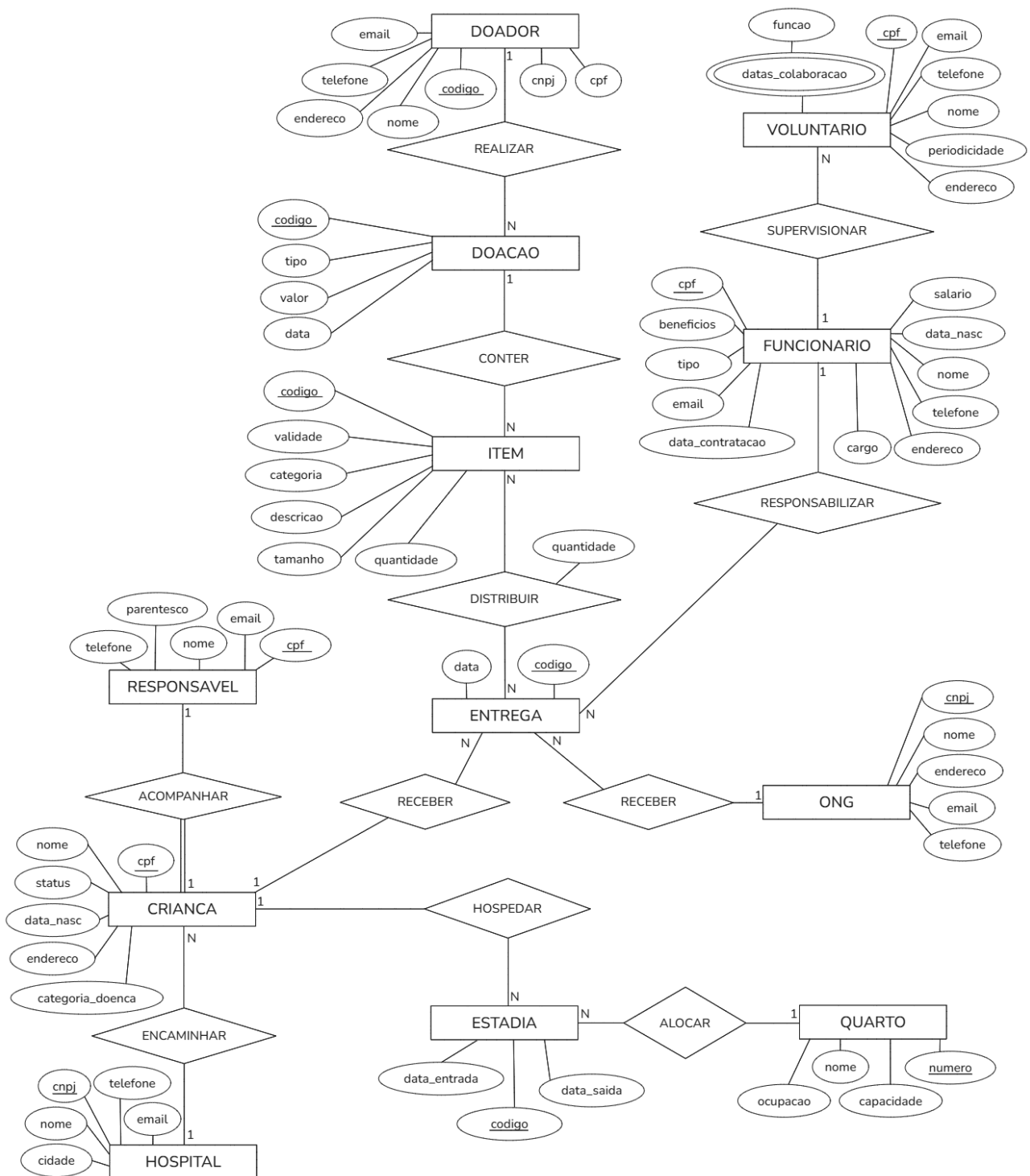
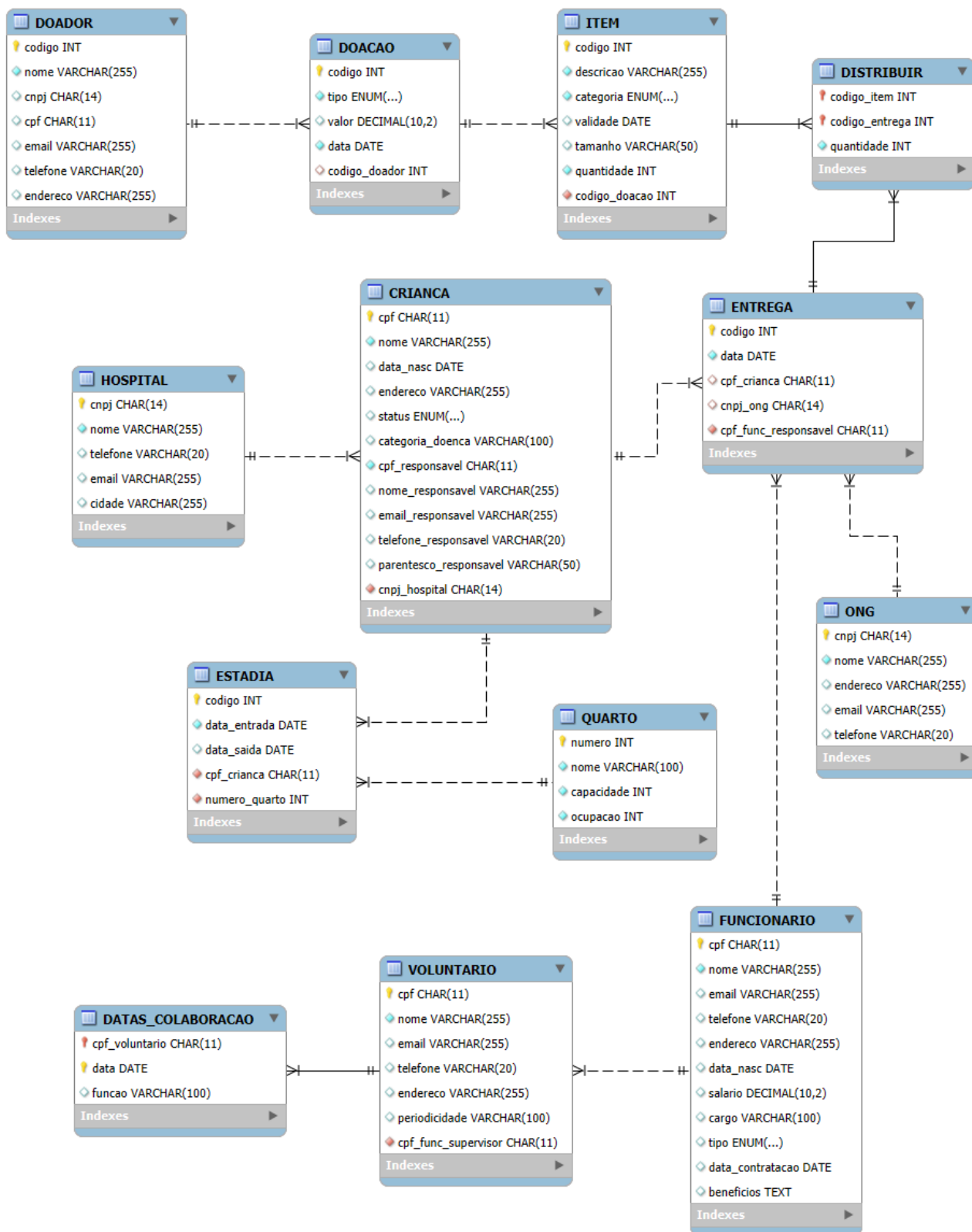


DIAGRAMA DE ESQUEMA BANCAPE



DIAGRAMA DE IMPLEMENTAÇÃO EM NOTAÇÃO PÉ-DE-GALINHA BANCAPE



SCRIPT SQL PARA CRIAÇÃO DO BANCO DE DADOS BANCAPE

```
-- Criando um banco de dados

CREATE DATABASE bancape;

-- Selecionando o banco de dados

USE bancape;


-- Drop tables if they exist to reset schema

DROP TABLE IF EXISTS DISTRIBUIR;

DROP TABLE IF EXISTS ESTADIA;

DROP TABLE IF EXISTS ITEM;

DROP TABLE IF EXISTS DOACAO;

DROP TABLE IF EXISTS DATAS_COLABORACAO;

DROP TABLE IF EXISTS VOLUNTARIO;

DROP TABLE IF EXISTS ENTREGA;

DROP TABLE IF EXISTS CRIANCA;

DROP TABLE IF EXISTS QUARTO;

DROP TABLE IF EXISTS FUNCIONARIO;

DROP TABLE IF EXISTS HOSPITAL;

DROP TABLE IF EXISTS ONG;

DROP TABLE IF EXISTS DOADOR;


-- DOADOR: pessoas físicas ou jurídicas

CREATE TABLE DOADOR (

    codigo INT AUTO_INCREMENT,

    nome VARCHAR(255) NOT NULL,

    cnpj CHAR(14),

    cpf CHAR(11),

    email VARCHAR(255),
```

```
telefone VARCHAR(20),

endereco VARCHAR(255),

PRIMARY KEY (codigo),

CHECK ( (cnpj IS NOT NULL AND cpf IS NULL) OR (cpf IS NOT NULL AND cnpj IS NULL) )

);
```

-- ONG colaboradora

```
CREATE TABLE ONG (

    cnpj CHAR(14) PRIMARY KEY,

    nome VARCHAR(255) NOT NULL,

    endereco VARCHAR(255),

    email VARCHAR(255),

    telefone VARCHAR(20)

);
```

-- HOSPITAL parceiro

```
CREATE TABLE HOSPITAL (

    cnpj CHAR(14) PRIMARY KEY,

    nome VARCHAR(255) NOT NULL,

    telefone VARCHAR(20),

    email VARCHAR(255),

    cidade VARCHAR(255)

);
```

-- QUARTO da hospedagem

```
CREATE TABLE QUARTO (

    numero INT PRIMARY KEY,

    nome VARCHAR(100) NOT NULL,
```

```
    capacidade INT NOT NULL CHECK (capacidade > 0),  
    ocupacao INT NOT NULL DEFAULT 0 CHECK (ocupacao >= 0 AND ocupacao <= capacidade)  
);
```

-- FUNCIONARIO com vínculo

```
CREATE TABLE FUNCIONARIO (  
    cpf CHAR(11) PRIMARY KEY,  
    nome VARCHAR(255) NOT NULL,  
    email VARCHAR(255),  
    telefone VARCHAR(20),  
    endereco VARCHAR(255),  
    data_nasc DATE,  
    salario DECIMAL(10,2),  
    cargo VARCHAR(100),  
    tipo ENUM('CLT','prestador','terceirizado'),  
    data_contratacao DATE,  
    beneficios TEXT  
);
```

-- CRIANCA e RESPONSÁVEL (mescados)

```
CREATE TABLE CRIANCA (  
    cpf CHAR(11) PRIMARY KEY,  
    nome VARCHAR(255) NOT NULL,  
    data_nasc DATE,  
    endereco VARCHAR(255),  
    status ENUM('ativo','alta','falecido'),  
    categoria_doenca VARCHAR(100),  
    cpf_responsavel CHAR(11) NOT NULL,
```



```
nome_responsavel VARCHAR(255),  
email_responsavel VARCHAR(255),  
telefone_responsavel VARCHAR(20),  
parentesco_responsavel VARCHAR(50),  
cnpj_hospital CHAR(14) NOT NULL,  
FOREIGN KEY (cnpj_hospital) REFERENCES HOSPITAL(cnpj)  
);
```

-- VOLUNTARIO sem vínculo empregatício

```
CREATE TABLE VOLUNTARIO (  
    cpf CHAR(11) PRIMARY KEY,  
    nome VARCHAR(255) NOT NULL,  
    email VARCHAR(255),  
    telefone VARCHAR(20),  
    endereco VARCHAR(255),  
    periodicidade VARCHAR(100),  
    cpf_func_supervisor CHAR(11) NOT NULL,  
    FOREIGN KEY (cpf_func_supervisor) REFERENCES FUNCIONARIO(cpf)  
);
```

-- DATAS_COLABORACAO (multivalorado)

```
CREATE TABLE DATAS_COLABORACAO (  
    cpf_voluntario CHAR(11),  
    data DATE,  
    funcao VARCHAR(100),  
    PRIMARY KEY (cpf_voluntario, data),  
    FOREIGN KEY (cpf_voluntario) REFERENCES VOLUNTARIO(cpf)  
);
```

-- DOACAO (monetária ou item)

```
CREATE TABLE DOACAO (  
    codigo INT AUTO_INCREMENT,  
    tipo ENUM('monetaria','item') NOT NULL,  
    valor DECIMAL(10,2),  
    data DATE NOT NULL,  
    codigo_doador INT,  
    PRIMARY KEY (codigo),  
    FOREIGN KEY (codigo_doador) REFERENCES DOADOR(codigo),  
    CHECK (  
        (tipo = 'monetaria' AND valor IS NOT NULL)  
        OR  
        (tipo = 'item' AND valor IS NULL)  
    )  
);
```

-- ITEM em estoque

```
CREATE TABLE ITEM (  
    codigo INT AUTO_INCREMENT,  
    descricao VARCHAR(255) NOT NULL,  
    categoria ENUM('roupas','alimentos','brinquedos','higiene','limpeza') NOT NULL,  
    validade DATE,  
    tamanho VARCHAR(50),  
    quantidade INT NOT NULL CHECK (quantidade >= 1),  
    codigo_doacao INT NOT NULL,  
    PRIMARY KEY (codigo),  
    FOREIGN KEY (codigo_doacao) REFERENCES DOACAO(codigo)
```

```
);
```

```
-- ENTREGA de itens aos destinatários
```

```
CREATE TABLE ENTREGA (  
    codigo INT AUTO_INCREMENT PRIMARY KEY,  
    data DATE NOT NULL,  
    cpf_crianca CHAR(11),  
    cnpj_ong CHAR(14),  
    cpf_func_responsavel CHAR(11) NOT NULL,  
    FOREIGN KEY (cpf_crianca) REFERENCES CRIANCA(cpf),  
    FOREIGN KEY (cnpj_ong) REFERENCES ONG(cnpj),  
    FOREIGN KEY (cpf_func_responsavel) REFERENCES FUNCIONARIO(cpf),  
    CHECK (  
        (cpf_crianca IS NOT NULL AND cnpj_ong IS NULL)  
        OR  
        (cpf_crianca IS NULL AND cnpj_ong IS NOT NULL)  
    )  
);
```

```
-- Relação N:N DISTRIBUIR entre ITEM e ENTREGA
```

```
CREATE TABLE DISTRIBUIR (  
    codigo_item INT,  
    codigo_entrega INT,  
    quantidade INT NOT NULL CHECK (quantidade > 0),  
    PRIMARY KEY (codigo_item, codigo_entrega),  
    FOREIGN KEY (codigo_item) REFERENCES ITEM(codigo),  
    FOREIGN KEY (codigo_entrega) REFERENCES ENTREGA(codigo)  
);
```

```

-- ESTADIA em quarto

CREATE TABLE ESTADIA (

    codigo INT AUTO_INCREMENT PRIMARY KEY,

    data_entrada DATE NOT NULL,

    data_saida DATE,

    cpf_crianca CHAR(11) NOT NULL,

    numero_quarto INT NOT NULL,

    FOREIGN KEY (cpf_crianca) REFERENCES CRIANCA(cpf),

    FOREIGN KEY (numero_quarto) REFERENCES QUARTO(numero),

    CHECK (data_saida IS NULL OR data_saida >= data_entrada)

);

```

SCRIPT SQL PARA POPULAÇÃO DO BANCO DE DADOS BANCAPE

```

-- Script para popular o banco de dados BANCAPE

-- Desabilita temporariamente as checagens de integridade referencial

SET FOREIGN_KEY_CHECKS = 0;


-- 1) DOADORES

INSERT INTO DOADOR (nome, cnpj, cpf, email, telefone, endereco)

VALUES

('João Silva', NULL, '12345678901', 'joao.silva@email.com', '(31)99999-0001', 'Rua das Flores, 100'),

('Empresa ABC Ltda', '12345678000190', NULL, 'contato@abc.com.br', '(31)3333-4444', 'Av. Brasil, 200');


-- 2) ONGs

INSERT INTO ONG (cnpj, nome, endereco, email, telefone)

VALUES

```

```
('98765432000165', 'ONG Amiga', 'Rua das Oliveiras, 50', 'contato@ongamiga.org',  
'(31)98888-1111'),  
  
( '55577788000122', 'ONG Parceira', 'Av. dos Anjos, 345', 'parceira@ong.org',  
'(31)97777-2222');
```

-- 3) HOSPITAIS

```
INSERT INTO HOSPITAL (cnpj, nome, telefone, email, cidade)
```

```
VALUES
```

```
('11122233000144', 'Hospital São Lucas', '(31)3277-8888', 'contato@saolucas.com.br',  
'Passos'),  
  
( '22233344000155', 'Hospital Bom Pastor', '(31)3344-5566', 'contato@bombastor.com.br',  
'Contagem');
```

-- 4) QUARTOS

```
INSERT INTO QUARTO (numero, nome, capacidade, ocupacao)
```

```
VALUES
```

```
(101, 'Beija-Flor', 2, 1),  
  
(102, 'Arara', 4, 2);
```

-- 5) FUNCIONÁRIOS

```
INSERT INTO FUNCIONARIO (
```

```
cpf, nome, email, telefone, endereco, data_nasc, salario, cargo, tipo, data_contratacao,  
beneficios
```

```
) VALUES
```

```
('22233344455', 'Maria Oliveira', 'maria.oliveira@cape.org', '(31)99999-2222', 'Av.  
Amazonas, 123, Belo Horizonte', '1980-05-10', 3500.00, 'Coordenadora', 'CLT',  
'2023-01-15', 'Plano de saúde'),  
  
( '33344455566', 'Carlos Santos', 'carlos.santos@cape.org', '(31)99998-3333', 'Rua do Sol,  
45, Belo Horizonte', '1975-09-20', 2800.00, 'Psicólogo', 'prestador', '2022-07-01',  
NULL);
```

-- 6) CRIANÇAS (com responsável mesclado)

```
INSERT INTO CRIANCA (
```

```
cpf, nome, data_nasc, endereco, status, categoria_doenca, cpf_responsavel,
nome_responsavel, email_responsavel, telefone_responsavel, parentesco_responsavel,
cnpj_hospital
```

```
) VALUES
```

```
('44455566677', 'Ana Beatriz Alves', '2015-03-12', 'Rua das Laranjeiras, 200, Araçuaí',
'ativo', 'hematologia', '55566677788', 'Pedro Alves', 'pedro.alves@email.com',
'(31)98877-3333', 'pai', '11122233000144'),
```

```
('55588899900', 'Lucas Coelho', '2018-11-05', 'Rua das Palmeiras, 400, Montes Claros',
'ativo', 'oncologia', '77788899911', 'Mariana Coelho', 'mariana.coelho@email.com',
'(31)98866-5555', 'mãe', '11122233000144');
```

```
-- 7) VOLUNTÁRIOS
```

```
INSERT INTO VOLUNTARIO (
```

```
cpf, nome, email, telefone, endereco, periodicidade, cpf_func_supervisor
```

```
) VALUES
```

```
('66677788899', 'Luisa Costa', 'luisa.costa@voluntaria.org', '(31)98800-4444', 'Rua das
Acácias, 300, Belo Horizonte', 'semanal', '22233344455');
```

```
-- 8) DATAS de COLABORAÇÃO dos VOLUNTÁRIOS
```

```
INSERT INTO DATAS_COLABORACAO (cpf_voluntario, data, funcao)
```

```
VALUES
```

```
('66677788899', '2025-06-01', 'assistente lúdico'),
```

```
('66677788899', '2025-06-15', 'apoio administrativo');
```

```
-- 9) DOAÇÕES
```

```
INSERT INTO DOACAO (tipo, valor, data, codigo_doador)
```

```
VALUES
```

```
('monetaria', 500.00, '2025-06-20', 1),
```

```
('item', NULL, '2025-06-21', 2);
```

```
-- 10) ITENS em ESTOQUE
```

```
INSERT INTO ITEM (
```

```

descricao, categoria, validade, tamanho, quantidade, codigo_doacao
) VALUES

('Camisas infantis', 'roupas', '2026-12-31', 'M', 10, 2),

('Brinquedos educativos', 'brinquedos', NULL, NULL, 5, 2);


-- 11) ENTREGAS

INSERT INTO ENTREGA (data, cpf_crianca, cnpj_ong, cpf_func_responsavel)

VALUES

('2025-06-25', '44455566677', NULL, '22233344455'),

('2025-06-26', NULL, '98765432000165', '33344455566');


-- 12) DISTRIBUIR (itens por entrega)

INSERT INTO DISTRIBUIR (codigo_item, codigo_entrega, quantidade)

VALUES

(1, 1, 2),

(2, 1, 1),

(2, 2, 2);


-- 13) ESTADIAS

INSERT INTO ESTADIA (data_entrada, data_saida, cpf_crianca, numero_quarto)

VALUES

('2025-06-22', '2025-06-28', '44455566677', 101),

('2025-06-23', NULL, '55588899900', 102);


-- Reabilita as checagens de integridade referencial

SET FOREIGN_KEY_CHECKS = 1;

```

CONSIDERAÇÕES FINAIS

Este foi o trabalho mais interessante que realizei na faculdade até agora, por unir a parte técnica com uma causa real e relevante. Ao invés de apenas resolver um problema fictício, tivemos a chance de pensar em soluções para uma situação concreta, que impacta diretamente a vida de muitas pessoas. Isso também nos levou a conhecer o trabalho de instituições não governamentais que atuam com seriedade e comprometimento para ajudar a sociedade. No meu caso, pude visitar a Casa de Acolhida Padre Eustáquio, um lugar incrível com uma causa muito nobre. A visita me ajudou a enxergar a importância desse tipo de iniciativa e a pensar em como um sistema pode de fato contribuir para sua organização e eficiência. Foi um trabalho que demonstrou que exatas e humanas não são opostas. Fiquei muito motivada a me engajar mais com causas sociais. Foi uma experiência enriquecedora, que uniu aprendizado técnico com propósito social, e que me marcou bastante. Obrigada, Wlad!

