



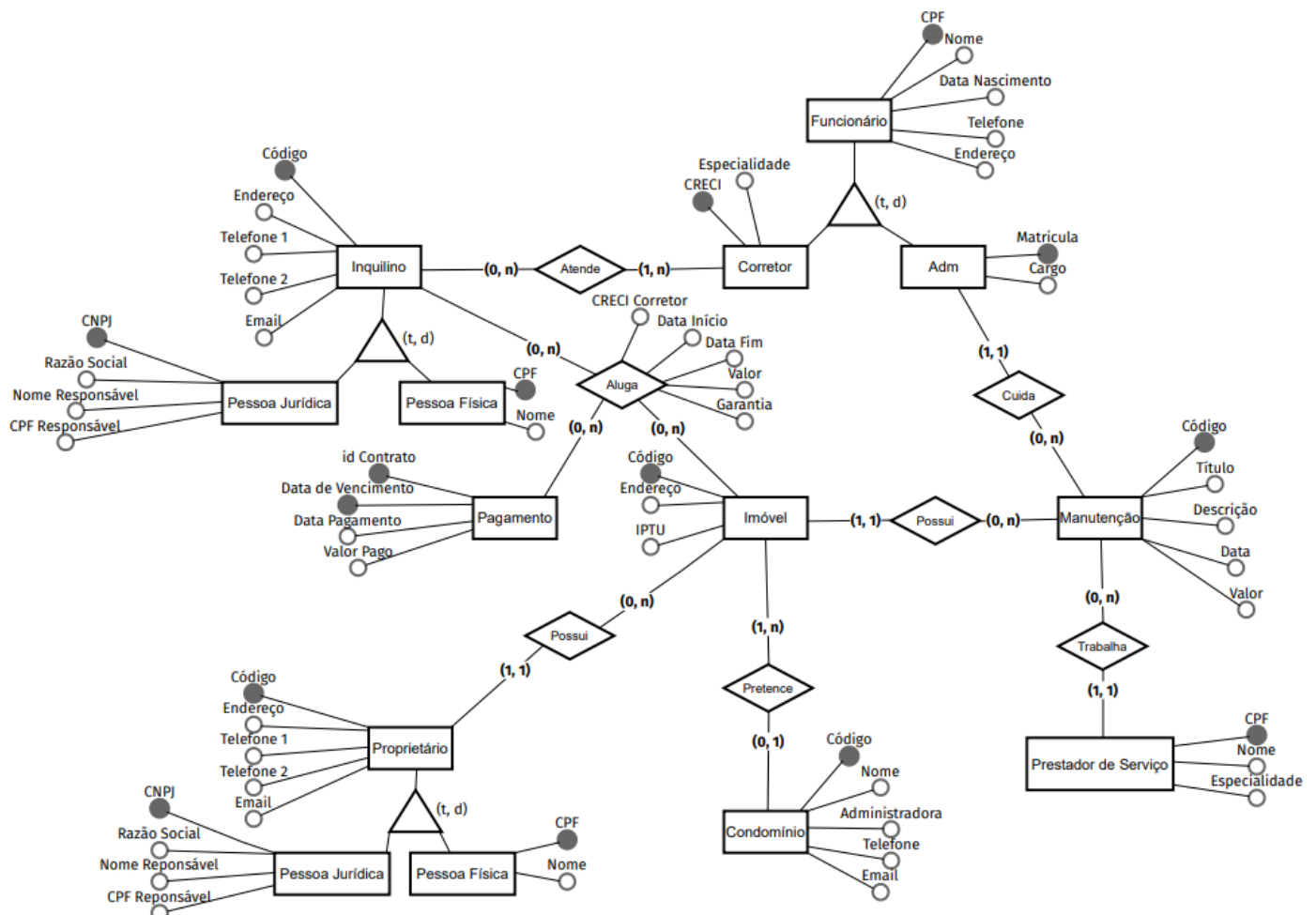
3ª AVALIAÇÃO

• DEFINIÇÃO DO PROBLEMA

BANCO DE DADOS DE IMOBILIÁRIA QUE TRABALHA COM LOCAÇÕES DE IMÓVEIS

Uma Imobiliária Regional trabalha com locações de imóveis e deseja um banco de dados para gerenciar suas informações. O banco de dados deve conter informações sobre funcionários (que podem ser corretores ou adm), inquilinos (que podem ser PF ou PJ), imóveis, proprietários (que podem ser PF ou PJ), condomínios, pagamentos de aluguel, manutenções e prestadores de serviços. Um corretor pode atender vários inquilinos e um inquilino pode ser atendido por um ou vários corretores. Um inquilino pode alugar vários imóveis cujos também podem ser alugados por vários inquilinos. Quando um inquilino aluga um imóvel são gerados os seguintes dados: CRECI do Corretor que realizou a locação, data de início, data de fim, valor e garantia, bem como as informações dos pagamentos de aluguel. Um imóvel pode pertencer a nenhum ou a um condomínio e pode ter apenas um proprietário, em um condomínio podem ter um ou vários imóveis e um proprietário pode possuir vários imóveis. Os imóveis podem precisar passar por nenhuma ou várias manutenções, mas uma manutenção pode ser feita em apenas um imóvel, estas manutenções devem ser cuidadas por um funcionário adm, e os funcionários adm podem cuidar de várias manutenções. Cada manutenção precisa ser realizada por um prestador de serviço, que por sua vez pode realizar várias manutenções.

• MODELO CONCEITUAL (DER)



- **MODELO LÓGICO RELACIONAL**

FUNCIONARIO (CPF, NOME, DATA_NASCIMENTO, TELEFONE, ENDERECO)
PK (CPF)

CORRETOR (CRECI, ESPECIALIDADE, CPF)
PK (CRECI)
FK (CPF) REFERENCIA FUNCIONARIO (CPF)

FUNCIONARIO_ADM (MATRICULA, CARGO, CPF)
PK (MATRICULA)
FK (CPF) REFERENCIA FUNCIONARIO (CPF)

INQUILINO (CODIGO_INQ, ENDEREÇO, TELEFONE_1, TELEFONE_2, EMAIL)
PK (CODIGO_INQ)

INQUILINO_PF (CPF, NOME, CODIGO_INQ)
PK (CPF)
FK (CODIGO_INQ) REFERENCIA INQUILINO (CODIGO_INQ)

INQUILINO_PJ (CNPJ, RAZAO_SOCIAL, CPF_RESP, NOME_RESP, CODIGO_INQ)
PK (CNPJ)
FK (CODIGO_INQ) REFERENCIA INQUILINO (CODIGO_INQ)

ATENDIMENTO (CRECI, COD_INQ)
PK (CRECI, COD_INQ)
FK (CRECI) REFERENCIA CORRETOR (CRECI)
FK (CODIGO_INQ) REFERENCIA INQUILINO (CODIGO_INQ)

PROPRIETARIO (CODIGO_PROP, ENDEREÇO, TELEFONE_1, TELEFONE_2, EMAIL)
PK (CODIGO_PROP)

PROPRIETARIO_PF (CPF, NOME, CODIGO_PROP)
PK (CPF)
FK (CODIGO_PROP) REFERENCIA PROPRIETARIO (CODIGO_PROP)

PROPRIETARIO_PJ (CNPJ, RAZAO_SOCIAL, CPF_RESP, NOME_RESP, CODIGO_PROP)
PK (CNPJ)
FK (CODIGO_PROP) REFERENCIA PROPRIETARIO (CODIGO_PROP)

CONDOMINIO (CODIGO_COND, NOME, ADMINISTRADORA, TELEFONE, EMAIL)
PK (CODIGO_COND)

IMOVEL (CODIGO_IM, ENDERECO, IPTU, COD_PROP, COD_COND)
PK (CODIGO_IM)
FK (CODIGO_PROP) REFERENCIA PROPRIETARIO (CODIGO_PROP)
FK (CODIGO_COND) REFERENCIA CONDOMINIO (CODIGO_COND)

ALUGUEL (ID_CONTRATO, CRECI_CORRETOR, DATA_INICIO, DATA_FIM, VALOR, GARANTIA, COD_INQ, COD_IM)
PK (ID_CONTRATO)
FK (COD_INQ) REFERENCIA INQUILINO (COD_INQ)
FK (COD_IM) REFERENCIA IMOVEL (CODIGO_IM)
FK (CRECI_CORRETOR) REFERENCIA CORRETOR (CRECI)

PAGAMENTOS (ID_CONTRATO, DATA_VENCIMENTO, DATA_PAGAMENTO, VALOR_PAGO)
PK (ID_CONTRATO, DATA_VENCIMENTO)
FK (ID_CONTRATO) REFERENCIA ALUGUEL (ID_CONTRATO)

PRESTADOR_SERVICO (CPF, NOME, ESPECIALIDADE)
PK (CPF)

MANUTENCAO (COD_MANUT, TITULO, DESCRICAO, DATA, VALOR, COD_IM, CPF, MATRICULA)
PK (COD_MANUT)
FK (COD_IM) REFERENCIA IMOVEL (CODIGO_IM)
FK (CPF) REFERENCIA PRESTADOR_SERVICO (CPF)
FK (MATRICULA) REFERENCIA FUNCIONARIO_ADM (MATRICULA)

- **MODELO FÍSICO (Sintaxe PostgreSQL)**

```
-- Tabela FUNCIONARIO
CREATE TABLE FUNCIONARIO (
  CPF INT PRIMARY KEY,
  NOME VARCHAR(100),
  DATA_NASCIMENTO DATE,
  TELEFONE VARCHAR(20),
  ENDERECO VARCHAR(200)
);
```

```
-- Tabela CORRETOR
CREATE TABLE CORRETOR (
  CRECI INT PRIMARY KEY,
  ESPECIALIDADE VARCHAR(100),
  CPF INT REFERENCES FUNCIONARIO(CPF)
);
```

```
-- Tabela FUNCIONARIO_ADM
CREATE TABLE FUNCIONARIO_ADM (
  MATRICULA INT PRIMARY KEY,
  CARGO VARCHAR(100),
  CPF INT REFERENCES FUNCIONARIO(CPF)
);
```

```
-- Tabela INQUILINO
CREATE TABLE INQUILINO (
  CODIGO_INQ INT PRIMARY KEY,
  ENDERECO VARCHAR(200),
  TELEFONE_1 VARCHAR(20),
  TELEFONE_2 VARCHAR(20),
  EMAIL VARCHAR(100)
);
```

```
-- Tabela INQUILINO_PF
CREATE TABLE INQUILINO_PF (
  CPF INT PRIMARY KEY,
  NOME VARCHAR(100),
  CODIGO_INQ INT REFERENCES INQUILINO(CODIGO_INQ)
);
```

```
-- Tabela INQUILINO_PJ
CREATE TABLE INQUILINO_PJ (
  CNPJ INT PRIMARY KEY,
  RAZAO_SOCIAL VARCHAR(100),
  CPF_RESP INT,
  NOME_RESP VARCHAR(100),
  CODIGO_INQ INT REFERENCES INQUILINO(CODIGO_INQ)
);
```

```

-- Tabela ATENDIMENTO
CREATE TABLE ATENDIMENTO (
  CRECI INT REFERENCES CORRETOR(CRECI),
  COD_INQ INT REFERENCES INQUILINO(CODIGO_INQ),
  PRIMARY KEY (CRECI, COD_INQ)
);

-- Tabela PROPRIETARIO
CREATE TABLE PROPRIETARIO (
  CODIGO_PROP INT PRIMARY KEY,
  ENDERECO VARCHAR(200),
  TELEFONE_1 VARCHAR(20),
  TELEFONE_2 VARCHAR(20),
  EMAIL VARCHAR(100)
);

-- Tabela PROPRIETARIO_PF
CREATE TABLE PROPRIETARIO_PF (
  CPF INT PRIMARY KEY,
  NOME VARCHAR(100),
  CODIGO_PROP INT REFERENCES PROPRIETARIO(CODIGO_PROP)
);

-- Tabela PROPRIETARIO_PJ
CREATE TABLE PROPRIETARIO_PJ (
  CNPJ INT PRIMARY KEY,
  RAZAO_SOCIAL VARCHAR(100),
  CPF_RESP INT,
  NOME_RESP VARCHAR(100),
  CODIGO_PROP INT REFERENCES PROPRIETARIO(CODIGO_PROP)
);

-- Tabela CONDOMINIO
CREATE TABLE CONDOMINIO (
  CODIGO_COND INT PRIMARY KEY,
  NOME VARCHAR(100),
  ADMINISTRADORA VARCHAR(100),
  TELEFONE VARCHAR(20),
  EMAIL VARCHAR(100)
);

-- Tabela IMOVEL
CREATE TABLE IMOVEL (
  CODIGO_IM INT PRIMARY KEY,
  ENDERECO VARCHAR(200),
  IPTU INT,
  COD_PROP INT REFERENCES PROPRIETARIO(CODIGO_PROP),
  COD_COND INT REFERENCES CONDOMINIO(CODIGO_COND)
);

-- Tabela ALUGUEL
CREATE TABLE ALUGUEL (
  ID_CONTRATO INT PRIMARY KEY,
  CRECI_CORRETOR INT REFERENCES CORRETOR(CRECI),
  DATA_INICIO DATE,
  DATA_FIM DATE,
  VALOR DECIMAL(10, 2),
  GARANTIA TEXT,
  COD_INQ INT REFERENCES INQUILINO(CODIGO_INQ),
  COD_IM INT REFERENCES IMOVEL(CODIGO_IM)
);

```

```

-- Tabela PAGAMENTOS
CREATE TABLE PAGAMENTOS (
  ID_CONTRATO INT REFERENCES ALUGUEL(ID_CONTRATO),
  DATA_VENCIMENTO DATE,
  DATA_PAGAMENTO DATE,
  VALOR_PAGO DECIMAL(10, 2),
  PRIMARY KEY (ID_CONTRATO, DATA_VENCIMENTO)
);

-- Tabela PRESTADOR_SERVICO
CREATE TABLE PRESTADOR_SERVICO (
  CPF INT PRIMARY KEY,
  NOME VARCHAR(100),
  ESPECIALIDADE VARCHAR(100)
);

-- Tabela MANUTENCAO
CREATE TABLE MANUTENCAO (
  COD_MANUT INT PRIMARY KEY,
  TITULO VARCHAR(100),
  DESCRICAO TEXT,
  DATA DATE,
  VALOR DECIMAL(10, 2),
  COD_IM INT REFERENCES IMOVEL(CODIGO_IM),
  CPF INT REFERENCES PRESTADOR_SERVICO(CPF),
  MATRICULA INT REFERENCES FUNCIONARIO_ADM(MATRICULA)
);

```

- **CONSULTA UTILIZANDO EXISTS: MOSTRE QUAIS IMÓVEIS ESTÃO ALUGADOS (Sintaxe PostgreSQL)**

```

SELECT *
FROM IMOVEL i
WHERE EXISTS (
  SELECT *
  FROM ALUGUEL a
  WHERE a.COD_IM = i.CODIGO_IM
);

```

- **CONSULTA UTILIZANDO HAVING: MOSTRE QUAL CORRETOR ALUGOU MAIS IMÓVEIS (CRECI E NOME) E A RESPECTIVA QUANTIDADE (Sintaxe PostgreSQL)**

```

SELECT c.CRECI, f.NOME, COUNT(*) AS total_alugueis
FROM CORRETOR c
JOIN ALUGUEL a ON c.CRECI = a.CRECI_CORRETOR
JOIN FUNCIONARIO f ON c.CPF = f.CPF
GROUP BY c.CRECI, f.NOME
HAVING COUNT(*) = (
  SELECT MAX(alugueis_count)
  FROM (
    SELECT COUNT(*) AS alugueis_count
    FROM ALUGUEL
    GROUP BY CRECI_CORRETOR
  ) AS count
);

```

- **FUNÇÃO: MOSTRAR OS CONTRATOS (ID_CONTRATO, IMÓVEL E INQUILINO) QUE POSSUEM PAGAMENTOS EM ATRASO (Sintaxe PostgreSQL)**

```
CREATE FUNCTION contratos_com_pagamentos_atrasados()
RETURNS TABLE (
    id_contrato INT,
    imovel INT,
    inquilino INT
) AS $$
BEGIN
    RETURN QUERY
    SELECT a.ID_CONTRATO, a.COD_IM, a.COD_INQ
    FROM ALUGUEL a
    INNER JOIN PAGAMENTOS p ON a.ID_CONTRATO = p.ID_CONTRATO
    WHERE p.DATA_VENCIMENTO < CURRENT_DATE AND p.DATA_PAGAMENTO IS NULL;
END;
$$ LANGUAGE plpgsql;
```

```
-- Consulta para rodar a função:
SELECT * FROM contratos_com_pagamentos_atrasados();
```

- **TRIGGER: ATUALIZAR O VALOR TOTAL DE MANUTENÇÕES EM UM IMÓVEL SEMPRE QUE UM NOVA MANUTENÇÃO FOR ADICIONADA (Sintaxe PostgreSQL)**

```
-- Adicionar a coluna VALOR_TOTAL_MANUTENCOES na tabela IMOVEL
ALTER TABLE IMOVEL
ADD COLUMN VALOR_TOTAL_MANUTENCOES DECIMAL(10, 2) DEFAULT 0;
```

```
-- Criar a trigger de atualização do valor total de manutenções em um imóvel
```

```
-- Criação da função
CREATE OR REPLACE FUNCTION calcular_valor_total_manutencoes()
RETURNS TRIGGER AS $$
DECLARE
    valor_total DECIMAL(10, 2);
BEGIN
    SELECT COALESCE(SUM(V valor), 0) INTO valor_total
    FROM MANUTENCAO
    WHERE COD_IM = NEW.COD_IM;
    UPDATE IMOVEL
    SET VALOR_TOTAL_MANUTENCOES = valor_total
    WHERE CODIGO_IM = NEW.COD_IM;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;
```

```
-- Criação da trigger
CREATE TRIGGER atualizar_valor_total_manutencoes
AFTER INSERT ON MANUTENCAO
FOR EACH ROW
EXECUTE FUNCTION calcular_valor_total_manutencoes();
```

- **ANEXO I – INSERTS NAS TABELAS PARA TESTAR AS CONSULTAS:**

-- Inserção na tabela FUNCIONARIO

```
INSERT INTO FUNCIONARIO (CPF, NOME, DATA_NASCIMENTO, TELEFONE, ENDERECO)
VALUES
```

```
(111111111, 'João Silva', '1990-01-01', '1234567890', 'Rua A, 123'),
(222222222, 'Maria Santos', '1985-05-10', '9876543210', 'Avenida B, 456'),
(333333333, 'Pedro Oliveira', '1995-09-15', '5555555555', 'Rua C, 789'),
(444444444, 'Ana Pereira', '1992-03-20', '9999999999', 'Avenida D, 987'),
(555555555, 'Lucas Souza', '1988-11-25', '7777777777', 'Rua E, 654');
```

-- Inserção na tabela CORRETOR

```
INSERT INTO CORRETOR (CRECI, ESPECIALIDADE, CPF)
VALUES
```

```
(111, 'Casas', 111111111),
(222, 'Apartamentos', 222222222),
(333, 'Galpões', 333333333),
(444, 'Salas Comerciais', 444444444),
(555, 'Residencial em Geral', 555555555);
```

-- Inserção na tabela FUNCIONARIO_ADM

```
INSERT INTO FUNCIONARIO_ADM (MATRICULA, CARGO, CPF)
VALUES
```

```
(1001, 'Gerente', 111111111),
(1002, 'Supervisor', 222222222),
(1003, 'Administrador', 333333333),
(1004, 'Coordenador', 444444444),
(1005, 'Supervisor', 555555555);
```

-- Inserção na tabela INQUILINO

```
INSERT INTO INQUILINO (CODIGO_INQ, ENDERECO, TELEFONE_1, TELEFONE_2, EMAIL)
VALUES
```

```
(10001, 'Rua X, 789', '111111111', '222222222', 'inquilino1@example.com'),
(10002, 'Avenida Y, 456', '333333333', '444444444', 'inquilino2@example.com'),
(10003, 'Rua Z, 123', '555555555', '666666666', 'inquilino3@example.com'),
(10004, 'Avenida W, 987', '777777777', '888888888', 'inquilino4@example.com'),
(10005, 'Rua V, 654', '999999999', '000000000', 'inquilino5@example.com');
```

-- Inserção na tabela INQUILINO_PF

```
INSERT INTO INQUILINO_PF (CPF, NOME, CODIGO_INQ)
VALUES
```

```
(111111111, 'João Silva', 10001),
(222222222, 'Maria Santos', 10002),
(333333333, 'Pedro Oliveira', 10003),
(444444444, 'Ana Pereira', 10004),
(555555555, 'Lucas Souza', 10005);
```

-- Inserção na tabela INQUILINO_PJ

```
INSERT INTO INQUILINO_PJ (CNPJ, RAZAO_SOCIAL, CPF_RESP, NOME_RESP, CODIGO_INQ)
VALUES
```

```
(123456789, 'Empresa A', 111111111, 'João Silva', 10001),
(987654321, 'Empresa B', 222222222, 'Maria Santos', 10002),
(456789123, 'Empresa C', 333333333, 'Pedro Oliveira', 10003),
(321654987, 'Empresa D', 444444444, 'Ana Pereira', 10004),
(789123456, 'Empresa E', 555555555, 'Lucas Souza', 10005);
```

-- Inserção na tabela ATENDIMENTO

```
INSERT INTO ATENDIMENTO (CRECI, COD_INQ)
VALUES
(111, 10001),
(222, 10002),
(333, 10003),
(444, 10004),
(555, 10005);
```

-- Inserção na tabela PROPRIETARIO

```
INSERT INTO PROPRIETARIO (CODIGO_PROP, ENDERECO, TELEFONE_1, TELEFONE_2, EMAIL)
VALUES
(20001, 'Rua M, 789', '111111111', '222222222', 'proprietario1@example.com'),
(20002, 'Avenida N, 456', '333333333', '444444444', 'proprietario2@example.com'),
(20003, 'Rua O, 123', '555555555', '666666666', 'proprietario3@example.com'),
(20004, 'Avenida P, 987', '777777777', '888888888', 'proprietario4@example.com'),
(20005, 'Rua Q, 654', '999999999', '000000000', 'proprietario5@example.com');
```

-- Inserção na tabela PROPRIETARIO_PF

```
INSERT INTO PROPRIETARIO_PF (CPF, NOME, CODIGO_PROP)
VALUES
(111111111, 'João Silva', 20001),
(222222222, 'Maria Santos', 20002),
(333333333, 'Pedro Oliveira', 20003),
(444444444, 'Ana Pereira', 20004),
(555555555, 'Lucas Souza', 20005);
```

-- Inserção na tabela PROPRIETARIO_PJ

```
INSERT INTO PROPRIETARIO_PJ (CNPJ, RAZAO_SOCIAL, CPF_RESP, NOME_RESP,
CODIGO_PROP)
VALUES
(987654321, 'Empresa F', 111111111, 'João Silva', 20001),
(123456789, 'Empresa G', 222222222, 'Maria Santos', 20002),
(789456123, 'Empresa H', 333333333, 'Pedro Oliveira', 20003),
(321654987, 'Empresa I', 444444444, 'Ana Pereira', 20004),
(654789321, 'Empresa J', 555555555, 'Lucas Souza', 20005);
```

-- Inserção na tabela CONDOMINIO

```
INSERT INTO CONDOMINIO (CODIGO_COND, NOME, ADMINISTRADORA, TELEFONE, EMAIL)
VALUES
(30001, 'Condomínio A', 'Administradora X', '111111111', 'condominio1@example.com'),
(30002, 'Condomínio B', 'Administradora Y', '222222222', 'condominio2@example.com'),
(30003, 'Condomínio C', 'Administradora Z', '333333333', 'condominio3@example.com'),
(30004, 'Condomínio D', 'Administradora W', '444444444', 'condominio4@example.com'),
(30005, 'Condomínio E', 'Administradora V', '555555555', 'condominio5@example.com');
```

-- Inserção na tabela IMOVEL

```
INSERT INTO IMOVEL (CODIGO_IM, ENDERECO, IPTU, COD_PROP, COD_COND)
VALUES
(40001, 'Rua S, 789', 1000, 20001, 30001),
(40002, 'Avenida T, 456', 2000, 20002, 30002),
(40003, 'Rua U, 123', 1500, 20003, 30003),
(40004, 'Avenida V, 987', 1800, 20004, 30004),
(40005, 'Rua W, 654', 1200, 20005, 30005);
```


-- Inserção na tabela ALUGUEL

```
INSERT INTO ALUGUEL (ID_CONTRATO, CRECI_CORRETOR, DATA_INICIO, DATA_FIM, VALOR,
GARANTIA, COD_INQ, COD_IM)
```

VALUES

```
(50001, 111, '2023-01-01', '2023-12-31', 1000.00, 'Caução', 10001, 40001),
(50002, 222, '2023-02-01', '2024-01-31', 1500.00, 'Fiador', 10002, 40002),
(50003, 333, '2023-03-01', '2024-02-29', 1200.00, 'Depósito', 10003, 40003),
(50004, 444, '2023-04-01', '2024-03-31', 1800.00, 'Seguro Fiança', 10004, 40004),
(50005, 555, '2023-05-01', '2024-04-30', 2000.00, 'Caução', 10005, 40005),
(50006, 555, '2023-06-01', '2024-05-30', 3000.00, 'Caução 2x', 10005, 40001);
```

-- Inserção na tabela PAGAMENTOS

```
INSERT INTO PAGAMENTOS (ID_CONTRATO, DATA_VENCIMENTO, DATA_PAGAMENTO,
VALOR_PAGO)
```

VALUES

```
(50001, '2023-01-05', '2023-01-05', 1000.00),
(50002, '2023-02-05', '2023-02-05', 1500.00),
(50003, '2023-03-05', '2023-03-05', 1200.00),
(50004, '2023-04-05', '2023-04-05', 1800.00),
(50005, '2023-05-05', '2023-05-05', 2000.00),
(50006, '2023-05-05', NULL, NULL);
```

-- Inserção na tabela PRESTADOR_SERVICO

```
INSERT INTO PRESTADOR_SERVICO (CPF, NOME, ESPECIALIDADE)
```

VALUES

```
(666666666, 'Fulano Prestador', 'Encanador'),
(777777777, 'Ciclano Serviços', 'Eletricista'),
(888888888, 'Beltrano Reparos', 'Pintor'),
(999999999, 'Empresa Serviços Gerais', 'Limpeza'),
(101010101, 'Outro Prestador', 'Diversos');
```

-- Inserção na tabela MANUTENCAO

```
INSERT INTO MANUTENCAO (COD_MANUT, TITULO, DESCRICAO, DATA, VALOR, COD_IM, CPF,
MATRICULA)
```

VALUES

```
(60001, 'Vazamento no banheiro', 'Reparo no encanamento do banheiro', '2023-01-10', 500.00, 40001,
666666666, 1001),
(60002, 'Problema na instalação elétrica', 'Conserto de fiação elétrica', '2023-02-15', 800.00, 40002,
777777777, 1002),
(60003, 'Pintura da sala', 'Pintura das paredes da sala', '2023-03-20', 600.00, 40003, 888888888, 1003),
(60004, 'Limpeza geral', 'Limpeza completa do imóvel', '2023-04-25', 400.00, 40004, 999999999, 1004),
(60005, 'Reparo na porta', 'Conserto da porta do quarto', '2023-05-30', 300.00, 40005, 101010101, 1005);
```