

➤ Criação das tabelas

```
SDROP TABLE IF EXISTS Regras CASCADE;
CREATE TABLE Regras (
    Regra TEXT PRIMARY KEY
);

DROP TABLE IF EXISTS Comodidades CASCADE;
CREATE TABLE Comodidades (
    Comodidade VARCHAR(100) PRIMARY KEY
);

DROP TABLE IF EXISTS Localizacao CASCADE;
CREATE TABLE Localizacao (
    CodigoLocal    NUMERIC          PRIMARY KEY,
    Cidade         VARCHAR(100)    NOT NULL,
    Estado         VARCHAR(100)    NOT NULL,
    Pais           VARCHAR(100)    NOT NULL,
    Bairro         VARCHAR(100)
);

DROP TABLE IF EXISTS Mensagem CASCADE;
CREATE TABLE Mensagem (
    Remetente      VARCHAR(100)    NOT NULL,
    Timestamp      TIMESTAMP       NOT NULL,
    Destinatario   VARCHAR(100)    NOT NULL,
    Texto          TEXT            NOT NULL,

    PRIMARY KEY (Remetente, Timestamp)
);

DROP TABLE IF EXISTS Usuario CASCADE;
CREATE TABLE Usuario (
    Nome           VARCHAR(100)    NOT NULL,
    Sobrenome      VARCHAR(100)    NOT NULL,
    Telefone       VARCHAR(20)     NOT NULL,
    DataNasc       DATE            NOT NULL,
    Endereco       VARCHAR(100)    NOT NULL,
    Sexo           VARCHAR(20)     NOT NULL,
    Email          VARCHAR(100)    NOT NULL,
    Senha          VARCHAR(100)    NOT NULL,
    CodigoLocal    NUMERIC          NOT NULL,
```

```

        PRIMARY KEY (Nome, Sobrenome, Telefone),
        FOREIGN KEY (CodigoLocal) REFERENCES
Localizacao(CodigoLocal)
);

DROP TABLE IF EXISTS Propriedade CASCADE;
CREATE TABLE Propriedade (
    NomeProp          VARCHAR(100)    NOT NULL,
    EnderecoProp      VARCHAR(100)    NOT NULL,
    Tipo              VARCHAR(100)    NOT NULL,
    NumQuartos        NUMERIC          NOT NULL,
    NumBanheiros      NUMERIC          NOT NULL,
    Pernoite          NUMERIC          NOT NULL,
    MaxHospedes       NUMERIC          NOT NULL,
    MinNoites         NUMERIC          NOT NULL,
    MaxNoites         NUMERIC          NOT NULL,
    DataDispInicial   Date,
    DataDispFinal     Date,
    TaxaLimpeza       NUMERIC,
    HCheckIn          TIME,
    HCheckOut         TIME,
    CodigoLocal       NUMERIC          NOT NULL,
    NomeAnfitriao     VARCHAR(100)    NOT NULL,
    SobrenomeAnfitriao VARCHAR(100)    NOT NULL,
    TelefoneAnfitriao VARCHAR(100)    NOT NULL,

    PRIMARY KEY (NomeProp, EnderecoProp),
    FOREIGN KEY (CodigoLocal) REFERENCES
Localizacao(CodigoLocal),
    FOREIGN KEY (NomeAnfitriao, SobrenomeAnfitriao,
TelefoneAnfitriao) REFERENCES Usuario(Nome, Sobrenome,
Telefone)
);

DROP TABLE IF EXISTS Locacao CASCADE;
CREATE TABLE Locacao (
    CodigoLoc          NUMERIC          PRIMARY KEY,
    NumHospedes        NUMERIC          NOT NULL,
    Preco              NUMERIC,
    Imposto            NUMERIC,
    PrecoTotal         NUMERIC,
    CodigoPromo        NUMERIC,
    DescontoPromo      NUMERIC,

```

```

Confirmada          BOOLEAN,
DataReserva         DATE          NOT NULL,
DataCheckIn         DATE          NOT NULL,
DataCheckOut        DATE          NOT NULL,
NomeLocatario       VARCHAR(100)  NOT NULL,
SobrenomeLocatario  VARCHAR(100)  NOT NULL,
TelefoneLocatario   VARCHAR(100)  NOT NULL,
NomePropriedade     VARCHAR(100)  NOT NULL,
EnderecoPropriedade VARCHAR(100)  NOT NULL,

    FOREIGN KEY (NomeLocatario, SobrenomeLocatario,
TelefoneLocatario) REFERENCES Usuario(Nome, Sobrenome,
Telefone),
    FOREIGN KEY (NomePropriedade, EnderecoPropriedade)
REFERENCES Propriedade(NomeProp, EnderecoProp)
);

DROP TABLE IF EXISTS PontosDeInteresse CASCADE;
CREATE TABLE PontosDeInteresse (
    PontoDeInteresse VARCHAR(200)  PRIMARY KEY,
    CodigoLocal       NUMERIC       NOT NULL,

    FOREIGN KEY (CodigoLocal) REFERENCES
Localizacao(CodigoLocal)
);

DROP TABLE IF EXISTS PropPossuiRegra CASCADE;
CREATE TABLE PropPossuiRegra (
    NomePropriedade     VARCHAR(100)  NOT NULL,
    EnderecoPropriedade VARCHAR(100)  NOT NULL,
    Regra               TEXT          NOT NULL,

    PRIMARY KEY (NomePropriedade, EnderecoPropriedade,
Regra),
    FOREIGN KEY (NomePropriedade, EnderecoPropriedade)
REFERENCES Propriedade(NomeProp, EnderecoProp),
    FOREIGN KEY (Regra) REFERENCES Regras(Regra)
);

DROP TABLE IF EXISTS PropPossuiComodidade CASCADE;
CREATE TABLE PropPossuiComodidade (
    NomePropriedade     VARCHAR(100)  NOT NULL,
    EnderecoPropriedade VARCHAR(100)  NOT NULL,

```

```

        Comodidade          VARCHAR(100)  NOT NULL,
        Quantidade          NUMERIC,

        PRIMARY KEY (NomePropriedade, EnderecoPropriedade,
Comodidade),
        FOREIGN KEY (NomePropriedade, EnderecoPropriedade)
REFERENCES Propriedade(NomeProp, EnderecoProp),
        FOREIGN KEY (Comodidade) REFERENCES
Comodidades(Comodidade)
);

DROP TABLE IF EXISTS ContaBancaria CASCADE;
CREATE TABLE ContaBancaria (
        NumeroConta          VARCHAR(100)  NOT NULL,
        NumeroRoteamento    VARCHAR(100)  NOT NULL,
        Tipo                  VARCHAR(100)  NOT NULL,
        NomeAnfitriao         VARCHAR(100)  NOT NULL,
        SobrenomeAnfitriao    VARCHAR(100)  NOT NULL,
        TelefoneAnfitriao     VARCHAR(20)   NOT NULL,

        PRIMARY KEY (NumeroConta, NomeAnfitriao,
SobrenomeAnfitriao, TelefoneAnfitriao),
        FOREIGN KEY (NomeAnfitriao, SobrenomeAnfitriao,
TelefoneAnfitriao) REFERENCES Usuario(Nome, Sobrenome,
Telefone)
);

DROP TABLE IF EXISTS Quarto CASCADE;
CREATE TABLE Quarto (
        Numero                NUMERIC      NOT NULL,
        QtdCamas              NUMERIC      NOT NULL,
        TipoCamas              VARCHAR(100)  NOT NULL,
        NomePropriedade        VARCHAR(100)  NOT NULL,
        EnderecoPropriedade    VARCHAR(100)  NOT NULL,

        PRIMARY KEY (Numero, NomePropriedade,
EnderecoPropriedade),
        FOREIGN KEY (NomePropriedade, EnderecoPropriedade)
REFERENCES Propriedade(NomeProp, EnderecoProp)
);

DROP TABLE IF EXISTS Avalia CASCADE;
CREATE TABLE Avalia (

```

NomePropriedade	VARCHAR(100)	NOT NULL,
EnderecoPropriedade	VARCHAR(100)	NOT NULL,
NomeLocatario	VARCHAR(100)	NOT NULL,
SobrenomeLocatario	VARCHAR(100)	NOT NULL,
TelefoneLocatario	VARCHAR(20)	NOT NULL,
RemetenteMensagem	VARCHAR(100)	NOT NULL,
TimestampMensagem	TIMESTAMP	NOT NULL,
NotaLocal	NUMERIC,	
NotaComunicacao	NUMERIC,	
NotaValor	NUMERIC,	
NotaLimpesza	NUMERIC,	
Fotos	VARCHAR(100),	

PRIMARY KEY (NomePropriedade, EnderecoPropriedade, NomeLocatario, SobrenomeLocatario, TelefoneLocatario, RemetenteMensagem, TimestampMensagem),

FOREIGN KEY (NomePropriedade, EnderecoPropriedade) REFERENCES Propriedade(NomeProp, EnderecoProp),

FOREIGN KEY (NomeLocatario, SobrenomeLocatario, TelefoneLocatario) REFERENCES Usuario(Nome, Sobrenome, Telefone),

FOREIGN KEY (RemetenteMensagem, TimestampMensagem) REFERENCES Mensagem(Remetente, Timestamp)

);

DROP TABLE IF EXISTS EnviaMensagem CASCADE;

CREATE TABLE EnviaMensagem (

NomeLocatario	VARCHAR(100)	NOT NULL,
SobrenomeLocatario	VARCHAR(100)	NOT NULL,
TelefoneLocatario	VARCHAR(100)	NOT NULL,
NomeAnfitriao	VARCHAR(100)	NOT NULL,
SobrenomeAnfitriao	VARCHAR(100)	NOT NULL,
TelefoneAnfitriao	VARCHAR(100)	NOT NULL,
RemetenteMensagem	VARCHAR(100)	NOT NULL,
TimestampMensagem	TIMESTAMP	NOT NULL,

PRIMARY KEY (NomeLocatario, SobrenomeLocatario, TelefoneLocatario, NomeAnfitriao, SobrenomeAnfitriao, TelefoneAnfitriao, RemetenteMensagem, TimestampMensagem),

FOREIGN KEY (NomeLocatario, SobrenomeLocatario, TelefoneLocatario) REFERENCES Usuario(Nome, Sobrenome, Telefone),

```
FOREIGN KEY (NomeAnfitriao, SobrenomeAnfitriao,
TelefoneAnfitriao) REFERENCES Usuario(Nome, Sobrenome,
Telefone),
FOREIGN KEY (RemetenteMensagem, TimestampMensagem)
REFERENCES Mensagem(Remetente, Timestamp)
);
```

- A requisição é trivial, basta utilizar o comando CREATE TABLE para cada relação mapeada, definindo seu nome e os atributos com seus devidos tipos. Note que a documentação detalhada sobre o conteúdo dessas tabelas encontra-se no documento “Documentação MRelacional”.
- Definiu-se as chaves primárias de cada relação, por meio da cláusula PRIMARY KEY, assim como as chaves estrangeiras, por meio das cláusulas FOREIGN KEY e REFERENCES (que especifica a tabela e as colunas às quais a chave estrangeira faz referência).
- É importante também atentar-se às restrições de integridade levantadas durante a criação do Modelo Relacional, a fim de garantir o bom funcionamento do banco de dados. Além disso, a obrigatoriedade de alguns atributos precisa ser levada em conta durante a criação das tabelas, justificando o uso da cláusula NOT NULL, por exemplo.