**Hackathon FUNETEC**

**Dicionário de Dados do Projeto**

**Instruções:**

Neste dicionário de dados um projeto por conter várias requisições, uma requisição pode conter muitas ordens de serviços. Uma ordem de serviço pode conter ou ou mais itens.

Um contrato pode ser associado somente a uma ordem de serviço, um contrato pode para pessoa jurídica ou pessoa física, se for para uma pessoa física, que tenha informado no tipoPessoa em contrato que 1 (PF), então deverá conter os lançamentos das parcelas, com controle de pagamentos recebidos.

**Relacionamentos e regras de negócio:**

* Um projeto → várias requisições.
* Uma requisição → várias ordens de serviço.
* Uma ordem → um ou mais itens.
* Um contrato → associado a **exatamente uma** ordem.
* Contrato com **tipoPessoa = 1 (PF)** possui lançamentos detalhados de **parcelas** e **controle de pagamentos** (itens\_contrato).

**1. Tabela de Projetos**

Nome: **projetos**

| Nome | Descrição | Chave | Tipo / Tamanho | Observação / Origem |
| --- | --- | --- | --- | --- |
| codProjeto | Código de identificação do projeto | Sim | Inteiro |  |
| nome | Nome do projeto |  | Caractere(200) |  |
| dataInicio | Data de início do projeto |  | Data | dd/mm/aaaa |
| dataEncerramento | Data prevista de finalização do projeto |  | Data | dd/mm/aaaa |
| valor | Valor orçado do projeto |  | Moeda |  |
| situacao | Situação do andamento do projeto |  | Caractere(20) | Valores de entrada:  1 - Aguardando Início  2 – Em andamento  3 – Paralisado  4 – Suspendo  5 – Cancelado  6 – Concluído |

**2. Tabela de Requisição**

Nome: **requisicao**

| Nome | Descrição | Chave | Tipo / Tamanho | Observação / Origem |
| --- | --- | --- | --- | --- |
| codRequisicao | Código da Requisição | Sim | Inteiro |  |
| codProjeto | Código de identificação do projeto | Sim | Inteiro | Tabela Projetos |
| descricao | Descrição da requisição |  | Caractere(500) |  |
| dataSolicitacao | Data de solicitação da requisição |  | Data | dd/mm/aaaa |
| dataLimite | Data limite de finalização do pedido requerido |  | Data | dd/mm/aaaa |
| valor | Valor da requisição |  | Moeda |  |
| situacao | Situação do andamento da requisição |  | Caractere(20) | Valores de entrada:  1 - Aguardando Início  2 – Em andamento  3 - Suspensa  4 – Cancelada  5 – Concluída |

**3. Tabela Ordem**

Nome: **ordem**

| Nome | Descrição | Chave | Tipo / Tamanho | Observação / Origem |
| --- | --- | --- | --- | --- |
| codOrdem | Número da ordem de serviço | Sim | Inteiro |  |
| codRequisicao | Código da Requisição | Sim | Inteiro | Tabela Requisição |
| descricao | Descrição da ordem de serviço |  | Caractere(500) |  |
| dataSolicitacao | Data de solicitação da ordem de serviço |  | Data | dd/mm/aaaa |
| dataLimite | Data limite de finalização da ordem de serviço |  | Data | dd/mm/aaaa |
| valor | Valor global da ordem de serviço |  | Moeda |  |
| situacao | Situação do andamento da ordem de serviço |  | Caractere(20) | Valores de entrada:  1 - Aguardando Início  2 – Em andamento  3 – Cancelada  4 – Concluída |

**4. Tabela Itens da Ordem**

Nome: **itens\_ordem**

| Nome | Descrição | Chave | Tipo / Tamanho | Observação / Origem |
| --- | --- | --- | --- | --- |
| codOrdem | Código da ordem de serviço | Sim | Inteiro | Tabela Ordem |
| codItem | Código do item da ordem de serviço | Sim | Inteiro |  |
| descricao | Descrição do item da ordem de serviço |  | Caractere(500) |  |
| dataSolicitacao | Data de solicitação da ordem de serviço |  | Data | dd/mm/aaaa |
| dataLimite | Data limite de entrega do item da ordem de serviço |  | Data | dd/mm/aaaa |
| valor | Valor do item da ordem de serviço |  | Moeda |  |
| dataRecebido | Data do recebimento do item |  | Data | dd/mm/aaaa |
| situacao | Situação do item na ordem de serviço |  | Caractere(20) | Valores de entrada:  1 - Lançado  2 – Cancelado  3 – Entregue |

**5. Tabela Contrato**

Nome: **contrato**

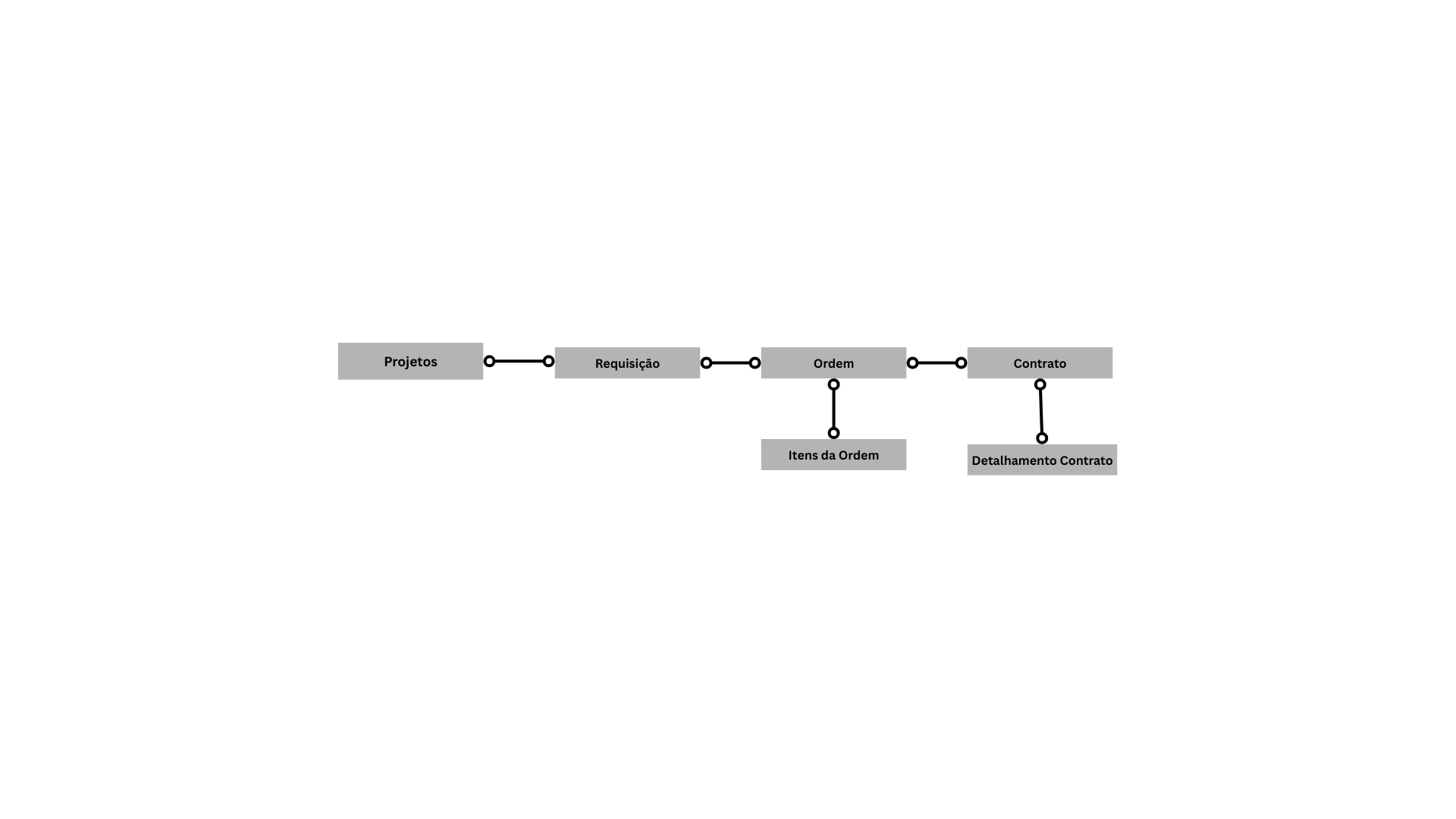
| Nome | Descrição | Chave | Tipo / Tamanho | Observação / Origem |
| --- | --- | --- | --- | --- |
| numContrato | Número do contrato | Sim | Caractere(10) | Número/ano formato: NNNN/AAAA |
| codOrdem | Código da ordem de serviço | Sim | Inteiro |  |
| descricao | Descrição do contrato |  | Caractere(500) |  |
| cpfcnpj | Número do CPF ou CNPJ do contratado |  | Caractere(14) |  |
| contratado | Nome do contratado |  | Caractere(150) |  |
| tipoPessoa | Tipo de pessoa contratada, se empresa ou pessoa física |  | Inteiro | 1 - Pessoa Física (PF)  2 - Pessoa Jurídica (PJ) |
| dataInicio | Data de início da vigência do contrato |  | Data | dd/mm/aaaa |
| dataFim | Data de finalização do contrato |  | Data | dd/mm/aaaa |
| valor | Valor do contrato |  | Moeda |  |
| parcelas | Quantidade de Parcelas |  | Inteiro |  |
| dataParcelaInicial | Data de vencimento da parcela inicial |  | Data | dd/mm/aaaa |
| situacao | Situação do andamento do contrato |  | Caractere(20) | Valores de entrada:  1 - Lançado  2 – Em andamento  3 – Cancelado  4 – Finalizado |

**6. Tabela Detalhamento do Contrato**

Nome: **itens\_contrato**

| Nome | Descrição | Chave | Tipo / Tamanho | Observação / Origem |
| --- | --- | --- | --- | --- |
| numContrato | Número do contrato | Sim | Caractere(10) | Tabela Contrato |
| codLancamento | Código de lançamento | Sim | Inteiro |  |
| dataLancamento | Data de lançamento do registro |  | Data | dd/mm/aaaa |
| numParcela | Número da Parcela lançada |  | Inteiro | Sequencial por contrato |
| valorParcela | Valor da parcela |  | Moeda |  |
| dataVencimento | Data de vencimento do pagamento da parcela |  | Data | dd/mm/aaaa |
| valorPago | Valor pago |  | Moeda |  |
| dataPagamento | Data de pagamento da parcela |  | Data | dd/mm/aaaa |
| situacao | Situação do andamento do contrato |  | Caractere(20) | Valores de entrada:  1 - Lançado  2 - Cancelado  3 - Liquidado |

Relacionamento entre as entidades de dados



**Script de criação do banco de dados em PostGresql:**

-- PROJETOS

CREATE TABLE IF NOT EXISTS projetos (

codProjeto INTEGER PRIMARY KEY,

nome VARCHAR(200) NOT NULL,

dataInicio DATE NOT NULL,

dataEncerramento DATE NOT NULL,

valor NUMERIC(14,2) NOT NULL,

situacao VARCHAR(20) NOT NULL CHECK (situacao IN ('1','2','3','4','5','6'))

);

-- REQUISICAO

CREATE TABLE IF NOT EXISTS requisicao (

codRequisicao INTEGER PRIMARY KEY,

codProjeto INTEGER NOT NULL,

descricao VARCHAR(500) NOT NULL,

dataSolicitacao DATE NOT NULL,

dataLimite DATE NOT NULL,

valor NUMERIC(14,2) NOT NULL,

situacao VARCHAR(20) NOT NULL CHECK (situacao IN ('1','2','3','4','5')),

CONSTRAINT fk\_requisicao\_projeto

FOREIGN KEY (codProjeto) REFERENCES projetos(codProjeto)

ON DELETE RESTRICT ON UPDATE CASCADE

);

-- ORDEM

CREATE TABLE IF NOT EXISTS ordem (

codOrdem INTEGER PRIMARY KEY,

codRequisicao INTEGER NOT NULL,

descricao VARCHAR(500) NOT NULL,

dataSolicitacao DATE NOT NULL,

dataLimite DATE NOT NULL,

valor NUMERIC(14,2) NOT NULL,

situacao VARCHAR(20) NOT NULL CHECK (situacao IN ('1','2','3','4')),

CONSTRAINT fk\_ordem\_requisicao

FOREIGN KEY (codRequisicao) REFERENCES requisicao(codRequisicao)

ON DELETE RESTRICT ON UPDATE CASCADE

);

-- ITENS\_ORDEM

CREATE TABLE IF NOT EXISTS itens\_ordem (

codOrdem INTEGER NOT NULL,

codItem INTEGER NOT NULL,

descricao VARCHAR(500) NOT NULL,

dataSolicitacao DATE NOT NULL,

dataLimite DATE NOT NULL,

valor NUMERIC(14,2) NOT NULL,

dataRecebido DATE,

situacao VARCHAR(20) NOT NULL CHECK (situacao IN ('1','2','3')),

CONSTRAINT pk\_itens\_ordem PRIMARY KEY (codOrdem, codItem),

CONSTRAINT fk\_itens\_ordem\_ordem

FOREIGN KEY (codOrdem) REFERENCES ordem(codOrdem)

ON DELETE CASCADE ON UPDATE CASCADE

);

-- CONTRATO (1:1 com ORDEM)

CREATE TABLE IF NOT EXISTS contrato (

numContrato VARCHAR(10) PRIMARY KEY,

codOrdem INTEGER NOT NULL UNIQUE,

descricao VARCHAR(500) NOT NULL,

cpfcnpj CHAR(14) NOT NULL CHECK (cpfcnpj ~ '^[0-9]{14}$'),

contratado VARCHAR(150) NOT NULL,

tipoPessoa INTEGER NOT NULL CHECK (tipoPessoa IN (1,2)), -- 1=PF, 2=PJ

dataInicio DATE NOT NULL,

dataFim DATE NOT NULL,

valor NUMERIC(14,2) NOT NULL,

parcelas INTEGER NOT NULL,

dataParcelaInicial DATE NOT NULL,

situacao VARCHAR(20) NOT NULL CHECK (situacao IN ('1','2','3','4')),

CONSTRAINT contrato\_num\_formato CHECK (numContrato ~ '^[0-9]{4}/[0-9]{4}$'),

CONSTRAINT fk\_contrato\_ordem

FOREIGN KEY (codOrdem) REFERENCES ordem(codOrdem)

ON DELETE RESTRICT ON UPDATE CASCADE

);

-- ITENS\_CONTRATO (lançamentos/parcelas)

CREATE TABLE IF NOT EXISTS itens\_contrato (

numContrato VARCHAR(10) NOT NULL,

codLancamento INTEGER NOT NULL,

dataLancamento DATE NOT NULL,

numParcela INTEGER NOT NULL,

valorParcela NUMERIC(14,2) NOT NULL,

dataVencimento DATE NOT NULL,

valorPago NUMERIC(14,2) NOT NULL DEFAULT 0,

dataPagamento DATE,

situacao VARCHAR(20) NOT NULL CHECK (situacao IN ('1','2','3')),

CONSTRAINT pk\_itens\_contrato PRIMARY KEY (numContrato, codLancamento),

CONSTRAINT fk\_itens\_contrato\_contrato

FOREIGN KEY (numContrato) REFERENCES contrato(numContrato)

ON DELETE CASCADE ON UPDATE CASCADE

);

-- Índices e regras adicionais

CREATE INDEX IF NOT EXISTS idx\_requisicao\_codProjeto ON requisicao(codProjeto);

CREATE INDEX IF NOT EXISTS idx\_ordem\_codRequisicao ON ordem(codRequisicao);

CREATE INDEX IF NOT EXISTS idx\_itens\_ordem\_situacao ON itens\_ordem(situacao);

CREATE INDEX IF NOT EXISTS idx\_contrato\_codOrdem ON contrato(codOrdem);

CREATE INDEX IF NOT EXISTS idx\_itens\_contrato\_status ON itens\_contrato(situacao);

CREATE INDEX IF NOT EXISTS idx\_itens\_contrato\_numPar ON itens\_contrato(numContrato, numParcela);

ALTER TABLE projetos ADD CONSTRAINT chk\_projetos\_periodo CHECK (dataEncerramento >= dataInicio);

ALTER TABLE requisicao ADD CONSTRAINT chk\_requisicao\_prazo CHECK (dataLimite >= dataSolicitacao);

ALTER TABLE ordem ADD CONSTRAINT chk\_ordem\_prazo CHECK (dataLimite >= dataSolicitacao);

ALTER TABLE itens\_ordem ADD CONSTRAINT chk\_itens\_ordem\_prazo CHECK (dataLimite >= dataSolicitacao);

ALTER TABLE itens\_contrato ADD CONSTRAINT chk\_itens\_contrato\_parcela CHECK (numParcela >= 1);

ALTER TABLE itens\_contrato ADD CONSTRAINT chk\_itens\_contrato\_liquidado\_valor CHECK ((situacao <> '3') OR (valorPago >= valorParcela));

**Script de criação do banco de dados em MySql:**

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

-- Schema: Projetos, Requisições, Ordens, Contratos

-- Banco: MySQL 8.x

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

-- Limpar tabelas na ordem de dependência

DROP TABLE IF EXISTS itens\_contrato;

DROP TABLE IF EXISTS contrato;

DROP TABLE IF EXISTS itens\_ordem;

DROP TABLE IF EXISTS ordem;

DROP TABLE IF EXISTS requisicao;

DROP TABLE IF EXISTS projetos;

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

-- 1) PROJETOS

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

CREATE TABLE projetos (

codProjeto INT PRIMARY KEY,

nome VARCHAR(200) NOT NULL,

dataInicio DATE NOT NULL,

dataEncerramento DATE NOT NULL,

valor DECIMAL(14,2) NOT NULL,

situacao ENUM('1','2','3','4','5','6') NOT NULL

-- 1=Aguardando Início, 2=Em andamento, 3=Paralisado, 4=Suspenso, 5=Cancelado, 6=Concluído

);

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

-- 2) REQUISICAO

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

CREATE TABLE requisicao (

codRequisicao INT PRIMARY KEY,

codProjeto INT NOT NULL,

descricao VARCHAR(500) NOT NULL,

dataSolicitacao DATE NOT NULL,

dataLimite DATE NOT NULL,

valor DECIMAL(14,2) NOT NULL,

situacao ENUM('1','2','3','4','5') NOT NULL,

CONSTRAINT fk\_requisicao\_projeto FOREIGN KEY (codProjeto)

REFERENCES projetos(codProjeto)

ON UPDATE CASCADE ON DELETE RESTRICT

);

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

-- 3) ORDEM

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

CREATE TABLE ordem (

codOrdem INT PRIMARY KEY,

codRequisicao INT NOT NULL,

descricao VARCHAR(500) NOT NULL,

dataSolicitacao DATE NOT NULL,

dataLimite DATE NOT NULL,

valor DECIMAL(14,2) NOT NULL,

situacao ENUM('1','2','3','4') NOT NULL,

CONSTRAINT fk\_ordem\_requisicao FOREIGN KEY (codRequisicao)

REFERENCES requisicao(codRequisicao)

ON UPDATE CASCADE ON DELETE RESTRICT

);

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

-- 4) ITENS\_ORDEM

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

CREATE TABLE itens\_ordem (

codOrdem INT NOT NULL,

codItem INT NOT NULL,

descricao VARCHAR(500) NOT NULL,

dataSolicitacao DATE NOT NULL,

dataLimite DATE NOT NULL,

valor DECIMAL(14,2) NOT NULL,

dataRecebido DATE,

situacao ENUM('1','2','3') NOT NULL,

PRIMARY KEY (codOrdem, codItem),

CONSTRAINT fk\_itens\_ordem\_ordem FOREIGN KEY (codOrdem)

REFERENCES ordem(codOrdem)

ON UPDATE CASCADE ON DELETE CASCADE

);

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

-- 5) CONTRATO (1:1 com ORDEM)

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

CREATE TABLE contrato (

numContrato VARCHAR(10) PRIMARY KEY,

codOrdem INT NOT NULL UNIQUE,

descricao VARCHAR(500) NOT NULL,

cpfcnpj CHAR(14) NOT NULL,

contratado VARCHAR(150) NOT NULL,

tipoPessoa TINYINT NOT NULL CHECK (tipoPessoa IN (1,2)), -- 1=PF, 2=PJ

dataInicio DATE NOT NULL,

dataFim DATE NOT NULL,

valor DECIMAL(14,2) NOT NULL,

parcelas INT NOT NULL,

dataParcelaInicial DATE NOT NULL,

situacao ENUM('1','2','3','4') NOT NULL,

CONSTRAINT fk\_contrato\_ordem FOREIGN KEY (codOrdem)

REFERENCES ordem(codOrdem)

ON UPDATE CASCADE ON DELETE RESTRICT

);

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

-- 6) ITENS\_CONTRATO (parcelas)

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

CREATE TABLE itens\_contrato (

numContrato VARCHAR(10) NOT NULL,

codLancamento INT NOT NULL,

dataLancamento DATE NOT NULL,

numParcela INT NOT NULL,

valorParcela DECIMAL(14,2) NOT NULL,

dataVencimento DATE NOT NULL,

valorPago DECIMAL(14,2) NOT NULL DEFAULT 0,

dataPagamento DATE,

situacao ENUM('1','2','3') NOT NULL,

PRIMARY KEY (numContrato, codLancamento),

CONSTRAINT fk\_itens\_contrato\_contrato FOREIGN KEY (numContrato)

REFERENCES contrato(numContrato)

ON UPDATE CASCADE ON DELETE CASCADE

);

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

-- Índices adicionais

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

CREATE INDEX idx\_requisicao\_codProjeto ON requisicao(codProjeto);

CREATE INDEX idx\_ordem\_codRequisicao ON ordem(codRequisicao);

CREATE INDEX idx\_itens\_ordem\_situacao ON itens\_ordem(situacao);

CREATE INDEX idx\_contrato\_codOrdem ON contrato(codOrdem);

CREATE INDEX idx\_itens\_contrato\_status ON itens\_contrato(situacao);

CREATE INDEX idx\_itens\_contrato\_numPar ON itens\_contrato(numContrato, numParcela);

# Observações sobre o banco de dados exemplo

# Volumetria mínima atendida

* Projetos: **500**
* Requisições: **1.000**
* Ordens: **1.500**
* Itens de Ordem: **variável (1–5 por OS)**
* Contratos: **~60%** das OS possuem contrato (**1:1** com OS; nenhuma OS tem mais de um contrato)
* Itens de Contrato (PF): **gerados apenas quando tipoPessoa = 1**, com distribuição de parcelas em **1, 2, 3, 4, 6, 8, 10 ou 12**, valores que **somam exatamente ao valor do contrato**, e **cenário de pagamento** aplicado: **80%** das parcelas pagas no **prazo**, **10%** pagas com **atraso** (até 15 dias) e **10%** **em aberto**.

# Observações técnicas

* Datas no formato **dd/mm/aaaa**
* Numeração de contrato: **NNNN/AAAA** (ex.: 0042/2025).
* Campos de **situação** seguem os **códigos do dicionário** (gravados como **texto de 1 dígito**), por tabela.
* Para **PF**, os lançamentos em itens\_contrato incluem **numParcela, dataVencimento, valorParcela, valorPago, dataPagamento** (quando aplicável) e **situacao** do lançamento fica **3 (liquidado)** quando quitado e **1 (lançado/parcial)** quando não quitado integralmente.
* Integridade garantida: **Projeto → Requisição (1:N)**, **Requisição → Ordem (1:N)**, **Ordem → Itens (1:N)**, **Ordem → Contrato (1:1)**, **Contrato → Itens\_Contrato (1:N)**.