	Ano Letivo: 2025	Turno: Noturno	Local: Presencial
	Curso: ENGENHARIA DA COMPUTAÇÃO		Turma: 3ª Fase
Disciplina: BANCO DE DADOS II		Professor(a): JORGE LUIZ DA SILVA	

TRABALHO FINAL COM BASE EM METODOLOGIAS ATIVAS DE APRENDIZAGEM

Projeto de banco de dados para um sistema de Locação/Aluguel de Equipamentos

Vanessa Ugioni ([vanessaugioni](#))

Karine Louis([karine285](#))

Sabrina Bettiol ([sabtl](#))

Ana Paula ([anapaulagesser](#))

Gabriel Rocha ([gabrieelrrocha](#))

Keller Hahan ([Keler1606](#))

Criciúma, 23/06/2025

URL do projeto no GitHub:

https://github.com/vanessaugioni/projeto_aluguel_equipamentos

The screenshot shows the GitHub repository page for 'projeto_aluguel_equipamentos' by user 'vanessaugioni'. The repository is public and has 0 stars, 0 forks, and 0 watches. The main branch is 'main'. The repository contains several files and folders, including 'crud-clientes', 'dicionario_dados', 'modelo_fisico', 'perguntas_negocio', 'relatorio_final', 'scripts', 'README.MD', 'image.png', and 'uso_CRUD.gif'. The repository also has 55 commits.

Repository Details:

- Repository: projeto_aluguel_equipamentos (Public)
- Owner: vanessaugioni
- Branches: 1 Branch (main)
- Tags: 0 Tags
- Commits: 55 Commits

Files and Folders:

Item	Commit Message	Time Ago
crud-clientes	detalhes-pergunta	1 hour ago
dicionario_dados	Add files via upload	19 hours ago
modelo_fisico	correcao-sql	2 hours ago
perguntas_negocio	ajuste-pergunta	50 minutes ago
relatorio_final	relatorio_final-versao2	1 hour ago
scripts	ajuste-pergunta	50 minutes ago
README.MD	Update README.MD	7 hours ago
image.png	ajustes-sql	3 hours ago
uso_CRUD.gif	readme-gif-crud	8 hours ago

Repository Statistics:

- Stars: 0
- Forks: 0
- Watching: 0

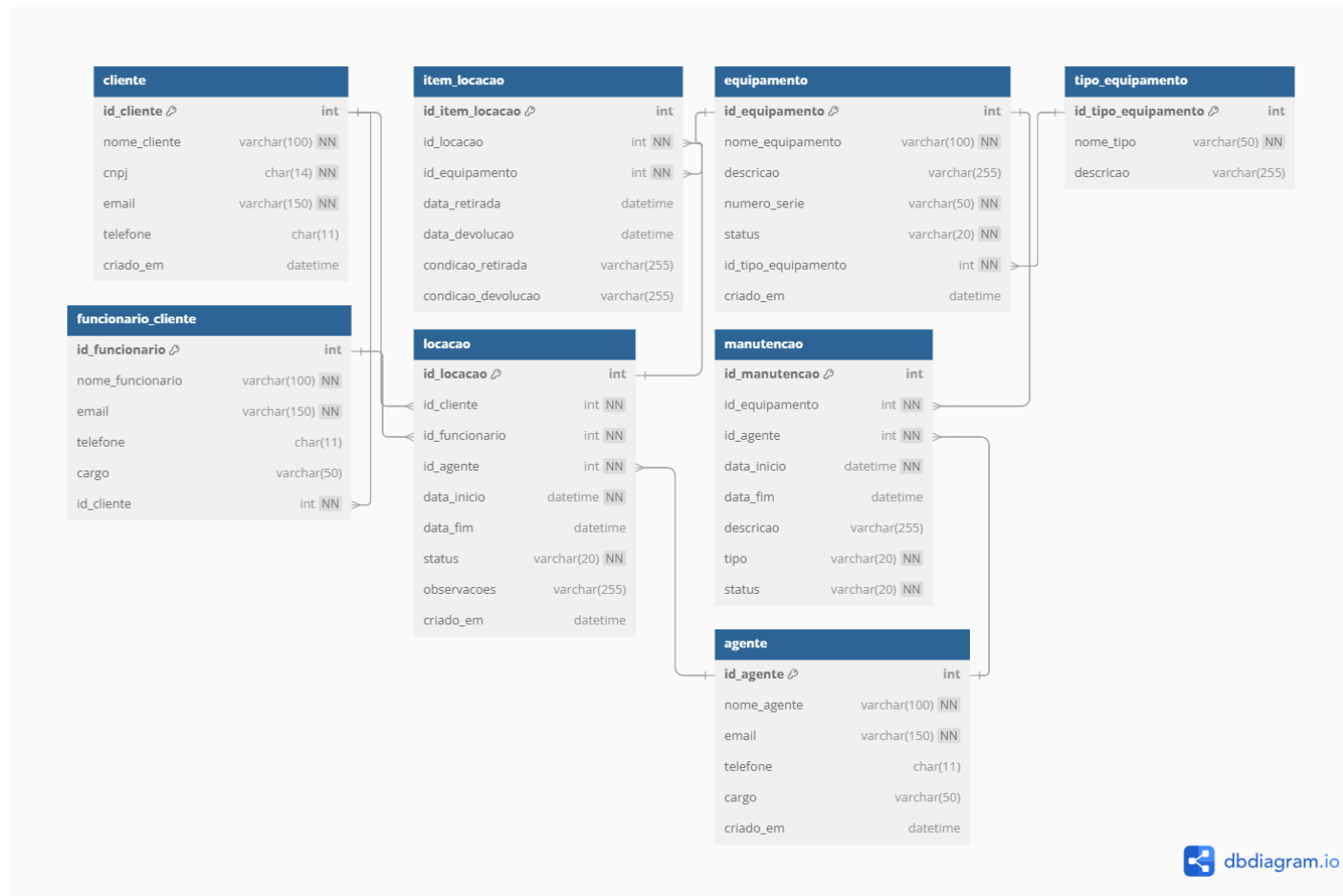
About: No description, website, or topics provided.

Releases: No releases published. [Create a new release](#)

Packages: No packages published. [Publish your first package](#)

Modelo ER Físico:

<https://dbdiagram.io/d/683e357061dc3bf08d3ad679>



Script dos comandos DDL para criação do Banco de dados (mínimo 8 tabelas)

```
-- =====  
-- 1. Table: cliente  
-- =====  
  
DROP TABLE IF EXISTS item_locacao;  
DROP TABLE IF EXISTS manutencao;  
DROP TABLE IF EXISTS locacao;  
DROP TABLE IF EXISTS equipamento;  
DROP TABLE IF EXISTS funcionario_cliente;  
DROP TABLE IF EXISTS agente;  
DROP TABLE IF EXISTS tipo_equipamento;  
DROP TABLE IF EXISTS cliente;  
  
GO  
  
CREATE TABLE cliente (  
    id_cliente INT PRIMARY KEY IDENTITY(1,1),  
    nome_cliente VARCHAR(100) NOT NULL,  
    cnpj CHAR(14) NOT NULL UNIQUE,  
    email VARCHAR(150) NOT NULL UNIQUE,  
    telefone CHAR(11) UNIQUE,  
    criado_em DATETIME  
);  
  
GO
```

```
-- =====
-- 2. Table: funcionario_cliente
-- =====

CREATE TABLE funcionario_cliente (
    id_funcionario INT PRIMARY KEY IDENTITY(1,1),
    nome_funcionario VARCHAR(100) NOT NULL,
    email VARCHAR(150) NOT NULL UNIQUE,
    telefone CHAR(11),
    cargo VARCHAR(50),
    id_cliente INT NOT NULL FOREIGN KEY REFERENCES cliente(id_cliente)
);
GO
```

```
-- =====
-- 3. Table: agente
-- =====

CREATE TABLE agente (
    id_agente INT PRIMARY KEY IDENTITY(1,1),
    nome_agente VARCHAR(100) NOT NULL,
    email VARCHAR(150) NOT NULL UNIQUE,
    telefone CHAR(11),
    cargo VARCHAR(50),
    criado_em DATETIME
);
GO
```

```
-- =====
```

```
-- 4. Table: tipo_equipamento
```

```
-- =====
```

```
CREATE TABLE tipo_equipamento (  
    id_tipo_equipamento INT PRIMARY KEY IDENTITY(1,1),  
    nome_tipo VARCHAR(50) NOT NULL,  
    descricao VARCHAR(255)  
);  
GO
```

```
-- =====
```

```
-- 5. Table: equipamento
```

```
-- =====
```

```
CREATE TABLE equipamento (  
    id_equipamento INT PRIMARY KEY IDENTITY(1,1),  
    nome_equipamento VARCHAR(100) NOT NULL,  
    descricao VARCHAR(255),  
    numeroSerie VARCHAR(50) NOT NULL UNIQUE,  
    status VARCHAR(20) NOT NULL DEFAULT 'Disponível',  
    id_tipo_equipamento INT NOT NULL FOREIGN KEY REFERENCES tipo_equipamento(id_tipo_equipamento),  
    criado_em DATETIME DEFAULT GETDATE(),  
    CONSTRAINT chk_status_equipamento CHECK (status IN ('Disponível', 'Indisponível'))  
);  
GO
```

```

-- =====
-- 6. Table: locacao
-- =====

CREATE TABLE locacao (
    id_locacao INT PRIMARY KEY IDENTITY(1,1),
    id_cliente INT NOT NULL FOREIGN KEY REFERENCES cliente(id_cliente),
    id_funcionario INT NOT NULL FOREIGN KEY REFERENCES funcionario_cliente(id_funcionario),
    id_agente INT NOT NULL FOREIGN KEY REFERENCES agente(id_agente),
    data_inicio DATETIME NOT NULL,
    data_fim DATETIME,
    status VARCHAR(20) NOT NULL,
    observacoes VARCHAR(255),
    criado_em DATETIME,
    CONSTRAINT chk_status_locacao CHECK (status IN ('Ativa', 'Finalizada', 'Cancelada'))
);
GO

```

```

-- =====
-- 7. Table: item_locacao
-- =====

CREATE TABLE item_locacao (
    id_item_locacao INT PRIMARY KEY IDENTITY(1,1),
    id_locacao INT NOT NULL FOREIGN KEY REFERENCES locacao(id_locacao),
    id Equipamento INT NOT NULL FOREIGN KEY REFERENCES equipamento(id_equipamento),
    data_retirada DATETIME,

```

```

        data_devolucao DATETIME,
        condicao_retirada VARCHAR(255),
        condicao_devolucao VARCHAR(255)
    );
GO

-- =====
-- 8. Table: manutencao
-- =====

CREATE TABLE manutencao (
    id_manutencao INT PRIMARY KEY IDENTITY(1,1),
    id_equipamento INT NOT NULL FOREIGN KEY REFERENCES equipamento(id_equipamento),
    id_agente INT NOT NULL FOREIGN KEY REFERENCES agente(id_agente),
    data_inicio DATETIME NOT NULL,
    data_fim DATETIME,
    descricao VARCHAR(255),
    tipo VARCHAR(20) NOT NULL,
    status VARCHAR(20) NOT NULL,
    CONSTRAINT chk_status_manutencao CHECK (status IN ('Pendente', 'Concluída', 'Cancelada'))
);
GO

```


Script que popula as tabelas do Banco de dados (mínimo 10 itens principais)

```
-- Populando cliente

SET IDENTITY_INSERT cliente ON

GO

INSERT cliente (id_cliente, nome_cliente, cnpj, email, telefone, criado_em) VALUES

(1, 'Tech Solutions LTDA', '12345678000101', 'contato@techsolutionsltda.com', '11951939783', GETDATE()),

(2, 'Construtora Ápice', '12345678000102', 'contato@construtoraapice.com', '11978321457', GETDATE()),

(3, 'Hospital São Lucas', '12345678000103', 'contato@hospitalsaolucas.com', '11962349811', GETDATE()),

(4, 'Universidade Pioneira', '12345678000104', 'contato@universidadepioneira.com', '11945632187', GETDATE()),

(5, 'Distribuidora Alfa', '12345678000105', 'contato@distribuidoraalfa.com', '11980976543', GETDATE()),

(6, 'Clínica Vida', '12345678000106', 'contato@clinicavida.com', '11971564892', GETDATE()),

(7, 'Loja Inova', '12345678000107', 'contato@lojainova.com', '11939485721', GETDATE()),

(8, 'Transportadora Rápida', '12345678000108', 'contato@transportadorarapida.com', '11982736459', GETDATE()),

(9, 'Fazenda Bela Vista', '12345678000109', 'contato@fazendabv.com', '11956473829', GETDATE()),

(10, 'Consultoria Prisma', '12345678000110', 'contato@consultoriaprisma.com', '11998653218', GETDATE()),

(11, 'Indústria Vértice', '12345678000111', 'contato@industriavertice.com', '11983512796', GETDATE())

GO

SET IDENTITY_INSERT cliente OFF

GO

-- Populando funcionario_cliente

SET IDENTITY_INSERT funcionario_cliente ON

GO

INSERT funcionario_cliente (id_funcionario, nome_funcionario, email, telefone, cargo, id_cliente) VALUES
```

```
(1, 'Lucas Silva', 'lucas.s@empresa.com', '11958010319', 'Consultor Comercial', 1),
(2, 'Mariana Almeida', 'mariana.a@empresa.com', '11968274103', 'Gerente de Contas', 2),
(3, 'Carlos Pereira', 'carlos.p@empresa.com', '11997314592', 'Analista de Projetos', 3),
(4, 'Fernanda Costa', 'fernanda.c@empresa.com', '11984927364', 'Suporte Técnico', 4),
(5, 'Ricardo Oliveira', 'ricardo.o@empresa.com', '11975830247', 'Consultor Comercial', 5),
(6, 'Juliana Mendes', 'juliana.m@empresa.com', '11993476281', 'Coordenador de TI', 6),
(7, 'André Souza', 'andre.s@empresa.com', '11964718392', 'Suporte Técnico', 7),
(8, 'Patrícia Lima', 'patricia.l@empresa.com', '11992018473', 'Analista de Projetos', 8),
(9, 'Bruno Rocha', 'bruno.r@empresa.com', '11983620947', 'Consultor Comercial', 9),
(10, 'Camila Martins', 'camila.m@empresa.com', '11977482936', 'Gerente de Contas', 10),
(11, 'Diego Santana', 'diego.s@empresa.com', '11986479123', 'Coordenador de TI', 11)
```

GO

```
SET IDENTITY_INSERT funcionario_cliente OFF
```

GO

```
-- Populando tipo_equipamento
```

```
SET IDENTITY_INSERT tipo_equipamento ON
```

GO

```
INSERT tipo_equipamento (id_tipo_equipamento, nome_tipo, descricao) VALUES
```

```
(1, 'Notebook', 'Computador portátil para tarefas gerais'),
(2, 'Monitor', 'Tela LCD/LED para computadores'),
(3, 'Impressora', 'Dispositivo para impressão de documentos'),
(4, 'Mouse', 'Periférico apontador'),
(5, 'Teclado', 'Periférico de entrada'),
```

```
(6, 'Switch de Rede', 'Equipamento de conexão em redes locais'),  
(7, 'Servidor', 'Equipamento de alto desempenho para serviços'),  
(8, 'Projetor', 'Dispositivo para projeção de imagem'),  
(9, 'Nobreak', 'Fonte de energia ininterrupta'),  
(10, 'Webcam', 'Câmera para videoconferência'),  
(11, 'Scanner', 'Equipamento para digitalização de documentos')
```

```
GO
```

```
SET IDENTITY_INSERT tipo_equipamento OFF
```

```
GO
```

```
-- Populando agente
```

```
SET IDENTITY_INSERT agente ON
```

```
GO
```

```
INSERT agente (id_agente, nome_agente, email, telefone, cargo, criado_em) VALUES
```

```
(1, 'Carlos Silva', 'carlos.silva@empresa.com', '11987654321', 'Técnico de Suporte', GETDATE()),  
(2, 'Fernanda Oliveira', 'fernanda.oliveira@empresa.com', '11987654322', 'Atendente', GETDATE()),  
(3, 'Marcos Pereira', 'marcos.pereira@empresa.com', '11987654323', 'Gerente de TI', GETDATE()),  
(4, 'Ana Souza', 'ana.souza@empresa.com', '11987654324', 'Técnico de Rede', GETDATE()),  
(5, 'Lucas Almeida', 'lucas.almeida@empresa.com', '11987654325', 'Atendente', GETDATE()),  
(6, 'Juliana Costa', 'juliana.costa@empresa.com', '11987654326', 'Gerente de Projetos', GETDATE()),  
(7, 'Rafael Gomes', 'rafael.gomes@empresa.com', '11987654327', 'Técnico de Hardware', GETDATE()),  
(8, 'Patrícia Lima', 'patricia.lima@empresa.com', '11987654328', 'Atendente', GETDATE()),  
(9, 'Thiago Martins', 'thiago.martins@empresa.com', '11987654329', 'Gerente de Suporte', GETDATE()),  
(10, 'Beatriz Fernandes', 'beatriz.fernandes@empresa.com', '11987654330', 'Técnico de Software', GETDATE()),
```

```
(11, 'Eduardo Rocha', 'eduardo.rocha@empresa.com', '11987654331', 'Atendente', GETDATE())

GO

SET IDENTITY_INSERT agente OFF

GO

-- Populando equipamento

SET IDENTITY_INSERT equipamento ON

GO

INSERT equipamento (id_equipamento, nome_equipamento, descricao, numero_serie, status, id_tipo_equipamento, criado_em) VALUES

(1, 'Lenovo ThinkPad X1', 'Notebook empresarial da Lenovo', 'SN0012LEN', 'Disponível', 1, '2025-06-11 10:00:00'),

(2, 'Asus VivoBook 15', 'Notebook Asus para uso geral', 'SN0013ASU', 'Disponível', 1, '2025-06-10 10:00:00'),

(3, 'HP EliteBook 840', 'Notebook de alta performance HP', 'SN0014HP', 'Disponível', 1, '2025-06-09 10:00:00'),

(4, 'MacBook Pro 14"', 'Notebook Apple para desenvolvedores', 'SN0015MAC', 'Disponível', 1, '2025-06-08 10:00:00'),

(5, 'Dell Latitude 7420', 'Notebook corporativo Dell', 'SN0016DEL', 'Disponível', 1, '2025-06-07 10:00:00'),

(6, 'Samsung Book i7', 'Notebook Samsung com bom custo-benefício', 'SN0017SAM', 'Disponível', 1, '2025-06-06 10:00:00'),

(7, 'Acer Aspire 5', 'Notebook Acer para tarefas diárias', 'SN0018ACE', 'Disponível', 1, '2025-06-05 10:00:00'),

(8, 'Positivo Motion Q', 'Notebook básico nacional', 'SN0019POS', 'Disponível', 1, '2025-06-04 10:00:00'),

(9, 'LG Gram 16', 'Notebook ultraleve LG', 'SN0020LG', 'Disponível', 1, '2025-06-03 10:00:00'),

(10, 'Microsoft Surface Laptop', 'Notebook Microsoft premium', 'SN0021MIC', 'Disponível', 1, '2025-06-02 10:00:00'),

(11, 'Aveall LIV A52', 'Notebook gamer brasileiro', 'SN0022AVE', 'Disponível', 1, '2025-06-01 10:00:00')

GO

SET IDENTITY_INSERT equipamento OFF

GO
```

```

-- Populando locacao

SET IDENTITY_INSERT locacao ON

GO

INSERT locacao (id_locacao, id_cliente, id_funcionario, id_agente, data_inicio, data_fim, status, observacoes, criado_em) VALUES
(1, 1, 1, 1, DATEADD(day, -10, GETDATE()), NULL, 'ativa', 'Locação para projeto temporário', GETDATE()),
(2, 2, 2, 2, DATEADD(day, -15, GETDATE()), DATEADD(day, -5, GETDATE()), 'finalizada', 'Equipamento devolvido em bom estado', GETDATE()),
(3, 3, 3, 3, DATEADD(day, -7, GETDATE()), NULL, 'ativa', 'Locação para evento corporativo', GETDATE()),
(4, 4, 4, 4, DATEADD(day, -30, GETDATE()), DATEADD(day, -1, GETDATE()), 'finalizada', 'Devolução com reparos necessários', GETDATE()),
(5, 5, 5, 5, DATEADD(day, -3, GETDATE()), NULL, 'ativa', 'Locação de equipamentos para home office', GETDATE()),
(6, 6, 6, 6, DATEADD(day, -20, GETDATE()), DATEADD(day, -10, GETDATE()), 'finalizada', 'Equipamento com defeito reportado', GETDATE()),
(7, 7, 7, 7, DATEADD(day, -5, GETDATE()), NULL, 'ativa', 'Locação para teste de software', GETDATE()),
(8, 8, 8, 8, DATEADD(day, -12, GETDATE()), DATEADD(day, -2, GETDATE()), 'finalizada', 'Locação concluída sem problemas', GETDATE()),
(9, 9, 9, 9, DATEADD(day, -1, GETDATE()), NULL, 'ativa', 'Equipamento para suporte remoto', GETDATE()),
(10, 10, 10, 10, DATEADD(day, -8, GETDATE()), NULL, 'ativa', 'Locação para treinamento interno', GETDATE()),
(11, 11, 11, 11, DATEADD(day, -18, GETDATE()), DATEADD(day, -3, GETDATE()), 'finalizada', 'Equipamento devolvido com acessórios',
GETDATE())

GO

SET IDENTITY_INSERT locacao OFF

GO

-- Populando item_locacao

SET IDENTITY_INSERT item_locacao ON

GO

INSERT item_locacao (id_item_locacao, id_locacao, id_equipamento, data_retirada, data_devolucao, condicao_retirada, condicao_devolucao)
VALUES
(1, 1, 1, DATEADD(day, -10, GETDATE()), NULL, 'Equipamento em perfeito estado', NULL),

```

```
(2, 2, 2, DATEADD(day, -15, GETDATE()), DATEADD(day, -5, GETDATE()), 'Novo', 'Funcionando corretamente'),
(3, 3, 3, DATEADD(day, -7, GETDATE()), NULL, 'Sem avarias', NULL),
(4, 4, 4, DATEADD(day, -30, GETDATE()), DATEADD(day, -1, GETDATE()), 'Leve desgaste', 'Pequenos reparos realizados'),
(5, 5, 5, DATEADD(day, -3, GETDATE()), NULL, 'Perfeito estado', NULL),
(6, 6, 6, DATEADD(day, -20, GETDATE()), DATEADD(day, -10, GETDATE()), 'Problemas no toner', 'Toner substituído'),
(7, 7, 7, DATEADD(day, -5, GETDATE()), NULL, 'Equipamento funcionando', NULL),
(8, 8, 8, DATEADD(day, -12, GETDATE()), DATEADD(day, -2, GETDATE()), 'Sem problemas', 'Devolução sem avarias'),
(9, 9, 9, DATEADD(day, -1, GETDATE()), NULL, 'Equipamento usado', NULL),
(10, 10, 10, DATEADD(day, -8, GETDATE()), NULL, 'Equipamento com acessórios completos', NULL),
(11, 11, 11, DATEADD(day, -18, GETDATE()), DATEADD(day, -3, GETDATE()), 'Bom estado geral', 'Devolvido com pequenas marcas')
```

GO

```
SET IDENTITY_INSERT item_locacao OFF
```

GO

-- Populando manutencao

```
SET IDENTITY_INSERT manutencao ON
```

GO

```
INSERT manutencao (id_manutencao, id_equipamento, id_agente, data_inicio, data_fim, descricao, tipo, status) VALUES
```

```
(1, 1, 2, '2025-05-30 10:00:00', NULL, 'Troca de teclado', 'Corretiva', 'Pendente'),
(2, 2, 3, '2025-05-28 10:00:00', NULL, 'Atualização de sistema', 'Corretiva', 'Pendente'),
(3, 3, 4, '2025-05-26 10:00:00', NULL, 'Substituição de HD', 'Corretiva', 'Concluída'),
(4, 4, 5, '2025-05-24 10:00:00', NULL, 'Manutenção em porta USB', 'Corretiva', 'Pendente'),
(5, 5, 6, '2025-05-22 10:00:00', NULL, 'Reinstalação do Windows', 'Corretiva', 'Pendente'),
(6, 6, 7, '2025-05-20 10:00:00', NULL, 'Substituição de bateria', 'Corretiva', 'Pendente'),
(7, 7, 8, '2025-05-18 10:00:00', NULL, 'Manutenção de tela', 'Corretiva', 'Concluída'),
```

```
(8, 8, 9, '2025-05-16 10:00:00', NULL, 'Limpeza de cooler', 'Corretiva', 'Pendente'),  
(9, 9, 10, '2025-05-14 10:00:00', NULL, 'Upgrade de memória RAM', 'Corretiva', 'Pendente'),  
(10, 10, 11, '2025-05-12 10:00:00', NULL, 'Troca de pasta térmica', 'Corretiva', 'Pendente'),  
(11, 11, 1, '2025-05-10 10:00:00', NULL, 'Verificação de desempenho', 'Corretiva', 'Concluída')  
GO  
  
SET IDENTITY_INSERT manutencao OFF  
  
GO
```

Principais Índices Utilizados no Modelo (Todos Não Clusterizados)

```
CREATE NONCLUSTERED INDEX idx_locacao_id_cliente ON locacao(id_cliente);  
  
CREATE NONCLUSTERED INDEX idx_locacao_id_funcionario ON locacao(id_funcionario);  
  
CREATE NONCLUSTERED INDEX idx_locacao_status_datafim ON locacao(status, data_fim);  
  
CREATE NONCLUSTERED INDEX idx_locacao_id_locacao_status ON locacao(id_locacao, status);  
  
  
  
  
CREATE NONCLUSTERED INDEX idx_item_locacao_id_locacao ON item_locacao(id_locacao);  
  
CREATE NONCLUSTERED INDEX idx_item_locacao_id Equipamento ON item_locacao(id_equipamento);  
  
CREATE NONCLUSTERED INDEX idx_item_locacao_datas ON item_locacao(data_retirada, data_devolucao);  
  
  
  
  
CREATE NONCLUSTERED INDEX idx_equipamento_tipo ON equipamento(id_tipo_equipamento);  
  
  
  
  
CREATE NONCLUSTERED INDEX idx_manutencao_data_idequipamento ON manutencao (data_inicio, id_equipamento);  
  
CREATE NONCLUSTERED INDEX idx_manutencao_status ON manutencao(status);  
  
  
  
  
CREATE NONCLUSTERED INDEX idx_funcionario_cliente_id_cliente ON funcionario_cliente(id_cliente);
```

Principais consultas mapeadas baseadas em regras de negócio (mínimo 6)

1. **Pergunta:** “ Qual é o ranking das 3 empresas que mais abriram manutenções nos últimos 6 meses, mostrando para cada uma a quantidade total de manutenções realizadas e a quantidade atual de manutenções pendentes?”

Aluno(a): Vanessa Ugioni (uso de Stored Procedure)

-- Plano de Execução - Atalho (ctrl + M)

```
CREATE OR ALTER PROCEDURE sp_ranking_manutencao_clientes
AS
BEGIN
    SET NOCOUNT ON;

    WITH manut_recntes AS (
        SELECT
            c.cnpj,
            c.nome_cliente,
            m.id_manutencao,
            m.status
        FROM manutencao m
        JOIN equipamento e ON m.id_equipamento = e.id_equipamento
        JOIN item_locacao il ON il.id_equipamento = e.id_equipamento
        JOIN locacao l ON l.id_locacao = il.id_locacao
        JOIN cliente c ON l.id_cliente = c.id_cliente
        WHERE m.data_inicio >= DATEADD(MONTH, -6, GETDATE())
    ),
```



```

contagem_manut AS (
    SELECT
        cnpj,
        nome_cliente,
        COUNT(*) AS total_manut_abertas,
        SUM(CASE WHEN status NOT IN ('Concluída', 'Cancelada') THEN 1 ELSE 0 END) AS manut_pendentes
    FROM manut_recentes
    GROUP BY cnpj, nome_cliente
)
SELECT TOP 3
    cnpj AS CNPJ_Empresa,
    nome_cliente AS Nome_Empresa,
    total_manut_abertas AS Total_Manutencoes_6M,
    manut_pendentes AS Manutencoes_Pendentes
FROM contagem_manut
ORDER BY total_manut_abertas DESC;
END;
GO

EXEC sp_ranking_manutencao_clientes;

```

/*

1. Plano de Acesso utilizado:

- A consulta usa predominantemente o operador Index Seek nos índices criados.
- Por exemplo, o índice idx_manutencao_data_idequipamento é utilizado para filtrar rapidamente as

manutenções dos últimos 6 meses pela coluna data_inicio e para buscar por id_equipamento.

2. Operadores usados:

- Index Seek: Busca eficiente em índices não clusterizados para filtrar por data e fazer junções.
- Nested Loops: Utilizado para realizar os joins entre tabelas (ex: entre manutencao e equipamento).
- Hash Match: Usado para agregações e operações de agrupamento na CTE.

3. Chaves dos índices utilizados:

- Para o índice idx_manutencao_data_idequipamento – as chaves usadas são as colunas (data_inicio, id_equipamento).
- Para idx_itemlocacao_idequipamento – a chave usada é id_equipamento.
- Para idx_locacao_idcliente – a chave usada é id_cliente.

*/

2. **Pergunta:** “Quais são os 5 equipamentos com maior número de locações no último ano, mostrando quantas vezes foram alugados e em quantas locações diferentes apareceram?”

Aluno(a): Sabrina Lorenzon Bettiol

*/

```
WITH locacoes_recentes AS (  
    SELECT  
        il.id_equipamento,  
        COUNT(*) AS total_vezes_alugado,  
        COUNT(DISTINCT il.id_locacao) AS total_locacoes_diferentes  
    FROM item_locacao il  
    JOIN locacao l ON il.id_locacao = l.id_locacao
```

```

        WHERE l.data_inicio >= DATEADD(YEAR, -1, GETDATE())
        GROUP BY il.id Equipamento
    )
SELECT TOP 5
    e.nome Equipamento,
    lr.total vezes alugado,
    lr.total locacoes diferentes
FROM locacoes_recentes lr
JOIN equipamento e ON lr.id Equipamento = e.id Equipamento
ORDER BY lr.total vezes alugado DESC;

```

-- Índices essenciais:

```

CREATE NONCLUSTERED INDEX idx_itemlocacao_idequipamento ON item_locacao(id Equipamento);
CREATE NONCLUSTERED INDEX idx_itemlocacao_idlocacao ON item_locacao(id_locacao);
CREATE NONCLUSTERED INDEX idx_locacao_datainicio ON locacao(data_inicio);

```

/*

Índices utilizados:

```

idx_locacao_datainicio = para filtrar o último ano usando locacao.data_inicio
idx_itemlocacao_idlocacao = JOIN entre item_locacao e locacao via id_locacao
idx_itemlocacao_idequipamento = GROUP BY em equipamento

```

Operadores principais:

Index Seek nos índices acima para busca eficiente

Nested Loop Join para combinar as tabelas item_locacao e locacao

Hash Match para agregação e contagem dos equipamentos alugados

Chaves de acesso:

item_locacao.id_equipamento

item_locacao.id_locacao

locacao.data_inicio

*/

3. **Pergunta:** “Quais são os 5 equipamentos mais alugados no último ano, informando a data da última manutenção e a média de dias alugados?”

Aluno(a): Ana Paula Bet Gesser

-- CTE para calcular a quantidade de dias alugados por equipamento

WITH CTE_Dias_Alugados AS (

SELECT

il.id_equipamento,

DATEDIFF(DAY, il.data_retirada, ISNULL(il.data_devolucao, GETDATE())) AS dias_alugados

FROM item_locacao il

INNER JOIN locacao l ON il.id_locacao = l.id_locacao

WHERE l.data_inicio >= DATEADD(YEAR, -1, GETDATE())

AND l.status IN ('Finalizada', 'Ativa')

),

-- CTE para média de dias por equipamento

```
CTE_Media_Dias AS (  
    SELECT  
        id_equipamento,  
        AVG(dias_alugados * 1.0) AS media_dias  
    FROM CTE_Dias_Alugados  
    GROUP BY id_equipamento  
) ,  
  
-- CTE para contar o número de locações por equipamento  
CTE_Quantidade_Locacoes AS (  
    SELECT  
        il.id_equipamento,  
        COUNT(*) AS total_locacoes  
    FROM item_locacao il  
    INNER JOIN locacao l ON l.id_locacao = il.id_locacao  
    WHERE l.data_inicio >= DATEADD(YEAR, -1, GETDATE())  
        AND l.status IN ('Finalizada', 'Ativa')  
    GROUP BY il.id_equipamento  
) ,  
  
-- CTE para pegar a última manutenção concluída  
CTE_Ultima_Manutencao AS (  
    SELECT  
        m.id_equipamento,  
        MAX(m.data_inicio) AS data_ultima_manutencao
```

```
FROM manutencao m
WHERE m.status = 'Concluída'
GROUP BY m.id Equipamento
)
```

-- Resultado final

```
SELECT TOP 5
    e.nome Equipamento,
    q.total Locacoes,
    um.data Ultima Manutencao,
    md.media_dias
FROM CTE_Quantidade_Locacoes q
JOIN equipamento e ON e.id Equipamento = q.id Equipamento
LEFT JOIN CTE_Media_Dias md ON md.id Equipamento = q.id Equipamento
LEFT JOIN CTE_Ultima_Manutencao um ON um.id Equipamento = q.id Equipamento
ORDER BY q.total Locacoes DESC;
```

Dicionário de Dados

Tabela	cliente					
Descrição	Armazena informações da empresa que está contratando a alocação dos equipamentos.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id_cliente	int		NOT NULL	X		Identificador único do cliente.
nome_cliente	varchar(100)	1-100	NOT NULL			Nome do cliente
cnpj	varchar(14)	14	NOT NULL			CNPJ do cliente (unico)
email	varchar(150)	6-150	NOT NULL			E-mail de contato (único).
telefone	varchar(11)	11				Número de telefone do cliente (opcional)
criado_em	datetime					Data de criação do registro
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
idx_cliente_pk	X		X	id_client		
idx_cliente_email		x	x	email		
idx_cliente_cnpj		x	x	cnpj		
Tabela	funcionario_cliente					
Descrição	Representa os funcionários da empresa cliente que irão utilizar ou gerenciar os equipamentos.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id_funcionario	int		NOT NULL	X		Identificador único do funcionário.

nome_funcionario	varchar(100)	1-100	NOT NULL			Nome completo
email	varchar(150)	6-150	NOT NULL			E-mail de contato (único).
telefone	varchar(11)	10-11				<u>Telefone para contato.</u>
cargo	varchar(50)	10-50				Cargo ou função desempenhada
id_cliente	int		NOT NULL		x	Relaciona o funcionário a um cliente (empresa contratante)
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
idx_funcionario_pk	x		x	id_funcionario		
idx_funcionario_cliente_email		x		email		
idx_funcionario_cliente_id_cliente		x		id_cliente		

Tabela	equipamento					
Descrição	Contém os dados dos equipamentos disponíveis para locação.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id_equipamento	int		NOT NULL	X		Identificador único do equipamento.
nome_equipamento	varchar(100)	1-100	NOT NULL			Nome do equipamento.
descricao	varchar(255)	1-225				Descrição do equipamento
numero_serie	varchar(50)	.1-50	NOT NULL			Número de série do equipamento (único).
status	varchar(20)	1-20.	NOT NULL			Situação atual (EX: Disponivel, etc...)
id_tipo_equipamento	int		NOT NULL		x	Tipo do equipamento
criado_em	datetime					Data de cadastro no sistema.
Índice						

Nome do índice	Clustered	NonClustered	Unique	Colunas
idx_equipamento	x		x	id_equipamento
idx_equipamento_numero_serie		x	x	numero_serie
idx_equipamento_id_tipo		x		id_tipo_equipamento

Tabela	tipo_equipamento
Descrição	Lista os tipos ou categorias dos equipamentos.

Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id_tipo_equipamento	int		NOT NULL	X		Identificador único do tipo.
nome_tipo	varchar(50)	1-50.	NOT NULL			Nome da categoria
descricao	varchar(255)	1-225				Descrição detalhada da categoria.

Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
idx_tipo_equipamento	x		x	id_tipo_equipamento		
idx_tipo_nome		x		nome_tipo		

Tabela	locacao
Descrição	Representa o processo de alocação dos equipamentos a um cliente.

Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição

id_locacao	int		NOT NULL	X		Identificador único da locação.
id_cliente	int		NOT NULL		x	Cliente que realizou a locação.
id_funcionario	int		NOT NULL		x	Funcionário que está vinculado à locação.
id_agente	int		NOT NULL		x	Funcionário da empresa de alocação responsável.
data_inicio	datetime		NOT NULL			Data de início da locação.
data_fim	datetime					Data prevista ou real de devolução.
status	varchar(20)	1-20.	NOT NULL			Situação da locação (Ex: cancelada, etc...)
observacoes	varchar(255)	1-225.				Observações adicionais.
criado_em	datetime					Data de criação do registro.
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
idx_locacao	x		x	id_locacao		
idx_locacao_id_cliente		x		id_cliente		
idx_locacao_id_funcionario		x		id_funcionario		
idx_locacao_id_agente				id_agente		
Tabela	item_locacao					
Descrição	Detalha os equipamentos específicos incluídos em uma locação.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id_item_locacao	int		NOT NULL	X		Identificador único do item.
id_locacao	int		NOT NULL		x	Qual locação esse item pertence.
id_equipamento	int		NOT NULL		x	Qual equipamento foi alocado.
data_retirada	datetime					Data de retirada do equipamento.

data_devolucao	datetime					Data de devolução.
condicao_retirada	varchar(255)	1-225				Condição do equipamento na retirada.
condicao_devolucao	varchar(255)	1-225				Condição na devolução.

Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
idx_id_item_locacao	x		x	id_item_locacao		
idx_item_locacao_id_locacao		x		id_locacao		
idx_item_locacao_id Equipamento		x		id Equipamento		

--	--

Tabela	manutencao
Descrição	Armazena as manutenções realizadas nos equipamentos.

Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id_manutencao	int		NOT NULL	X		Identificador único da manutenção.
id Equipamento	int		NOT NULL		x	Equipamento que passou por manutenção.
id_agente	int		NOT NULL		x	Funcionário técnico responsável.
data_inicio	datetime		NOT NULL			Data de início da manutenção.
data_fim	datetime					Data de conclusão.
descricao	varchar(255)	1-225				Detalhes do que foi feito.
tipo	varchar(20)	1-20.	NOT NULL			Tipo de manutenção (preventiva ou corretiva).
status	varchar(20)	1-20.	NOT NULL			Situação (em andamento, concluída).

Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		

idx_id_manutencao	x		x	id_manutencao
idx_manutencao_id Equipamento		x		id Equipamento
idx_manutencao_id Agente		x		id Agente

Tabela	agente					
Descrição	Profissionais da empresa de alocação que gerenciam locações ou realizam manutenções.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id_agente	int		NOT NULL	X		Identificador único do agente.
nome_agente	varchar(100)	1-100	NOT NULL			Nome do responsável.
email	varchar(150)	1-150	NOT NULL			E-mail para contato (único).
telefone	char(11)	11				Telefone
cargo	varchar(50)	1-50.				Cargo
criado_em	datetime					Data de criação do registro.
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
idx_agente_pk	x		x	id_agente		
idx_agente_email		x	x	email		

