**Centro Universitário UNISATC**

Engenharia de Software 3a fase – Banco de Dados II – Prof.Jorge Luiz da Silva

**TRABALHO FINAL COM BASE EM METODOLOGIAS ATIVAS DE APRENDIZAGEM**

**Projeto de banco de dados para um sistema de Aluguel de Carros**

**Arturo Burigo -** [**https://github.com/arturoburigo**](https://github.com/arturoburigo)

**Edrik Steiner -** [**https://github.com/edrikfsteiner**](https://github.com/edrikfsteiner)

**Gustavo Niero -** [**https://github.com/gustavoniero**](https://github.com/gustavoniero)

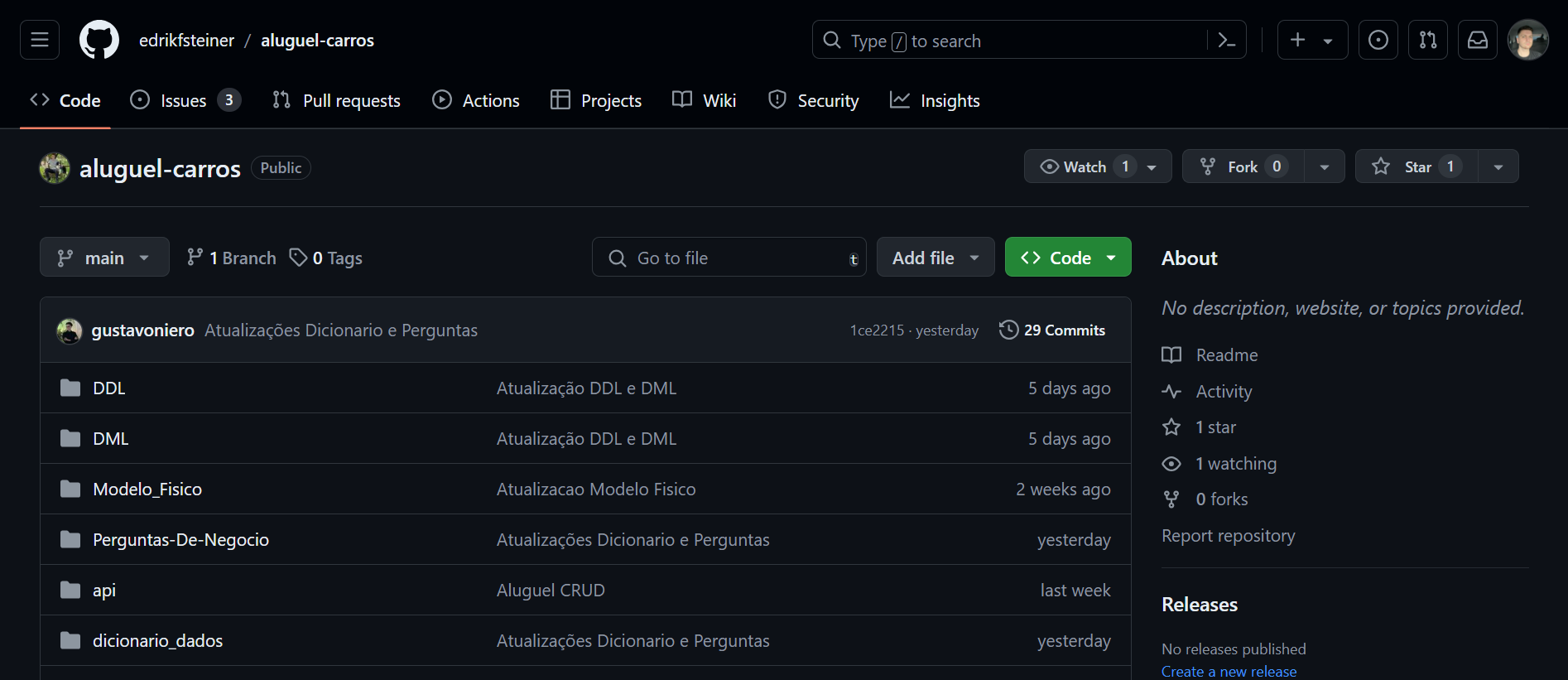
**Luiz Antônio Bezerra -** [**https://github.com/BezerraLuiz**](https://github.com/BezerraLuiz)

**Vitor Wessler -** [**https://github.com/VitorKW**](https://github.com/VitorKW)

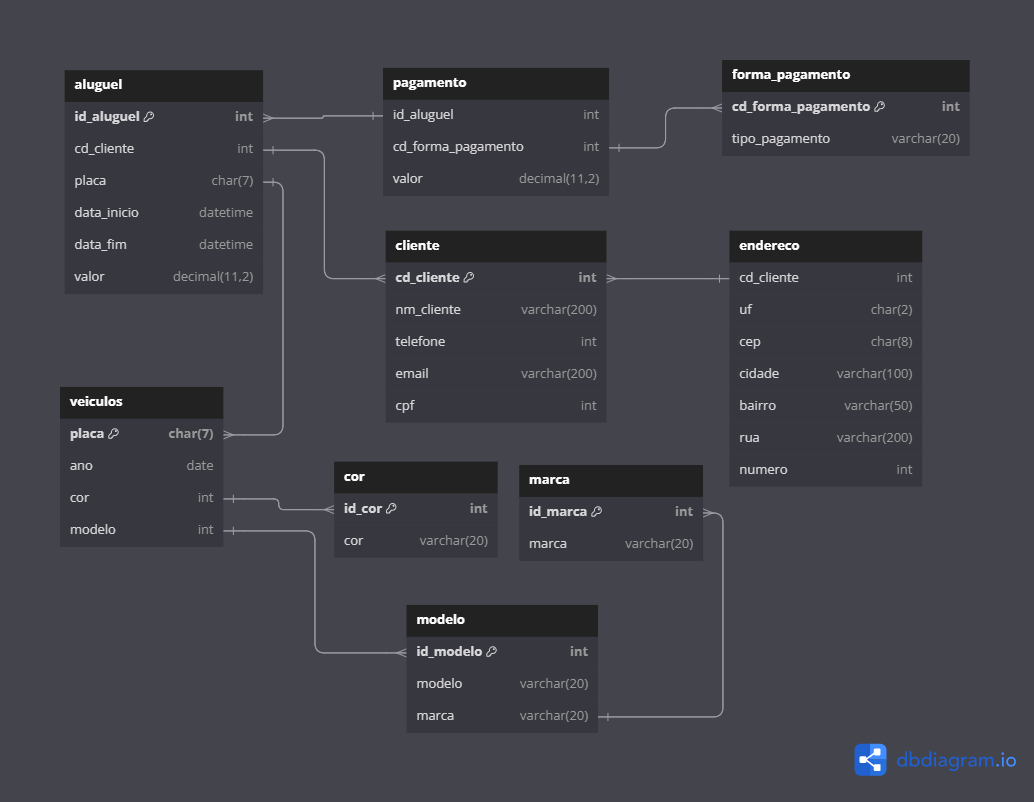
Criciúma, 25/06/2024

**URL do projeto no GitHub**

https://github.com/edrikfsteiner/aluguel-carros



**Modelo ER Físico**



**Dicionário de Dados**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tabela** | Marca | | | | | | |
| **Descrição** | Tabela responsável por armazenar as marcas dos carros | | | | | | |
| **Atributos** | | | | | | | |
| **Nome da Coluna** | | **Tipo do Dado** | **Valor min e max** | **Nulidade** | **PK** | **FK** | **Descrição** |
| id\_marca | | int | 1 - 2147483647 | NOT NULL | X |  | Código da marca |
| marca | | varchar(20) | 1 e 20 | NOT NULL |  |  | Nome da marca do veículo |
| **Índice** | | | | | | | |
| **Nome do índice** | | **Clustered** | **NonClustered** | **Unique** | **Colunas** | | |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **Tabela** | Modelo | | | | | | |
| **Descrição** | Tabela responsável por armazenar os modelos dos carros | | | | | | |
| **Atributos** | | | | | | | |
| **Nome da Coluna** | | **Tipo do Dado** | **Valor min e max** | **Nulidade** | **PK** | **FK** | **Descrição** |
| id\_modelo | | int | 1 - 2147483647 | NOT NULL | X |  | Código do modelo |
| modelo | | varchar(20) | 1 e 20 | NOT NULL |  |  | Modelo do Veículo |
| id\_marca | | int | 1 - 2147483647 | NOT NULL |  | X | Referência da tabela marca |
| **Índice** | | | | | | | |
| **Nome do índice** | | **Clustered** | **NonClustered** | **Unique** | **Colunas** | | |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **Tabela** | Cliente | | | | | | |
| **Descrição** | Tabela responsável por armazenar os dados dos Clientes | | | | | | |
| **Atributos** | | | | | | | |
| **Nome da Coluna** | | **Tipo do Dado** | **Valor min e max** | **Nulidade** | **PK** | **FK** | **Descrição** |
| cd\_cliente | | int | 1 - 2147483647 | NOT NULL | X |  | Código do Cliente |
| nm\_cliente | | varchar(200) | 1 e 200 | NOT NULL |  |  | Nome do Cliente |
| telefone | | varchar(15) | 1 e 15 | NOT NULL |  |  | Telefone do Cliente |
| email | | varchar(200) | 1 e 200 | NOT NULL |  |  | E-mail do Clientes |
| cpf | | char(11) | 11 e 11 | NOT NULL |  |  | CPF do Cliente |
| id\_endereco | | int | 1 - 2147483647 | NOT NULL |  | X | Referência da tabela endereço |
| **Índice** | | | | | | | |
| **Nome do índice** | | **Clustered** | **NonClustered** | **Unique** | **Colunas** | | |
| idx\_cliente | | X |  |  | cliente(cd\_cliente) | | |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **Tabela** | Endereço | | | | | | |
| **Descrição** | Tabela responsável por os endereços dos clientes | | | | | | |
| **Atributos** | | | | | | | |
| **Nome da Coluna** | | **Tipo do Dado** | **Valor min e max** | **Nulidade** | **PK** | **FK** | **Descrição** |
| id\_endereco | | int | 1 - 2147483647 | NOT NULL | X |  | Código do Endereço |
| rua | | varchar(200) | 1 e 200 | NOT NULL |  |  | Rua do Cliente |
| numero | | int | 1 - 2147483647 | NOT NULL |  |  | Número da Residência do Cliente |
| bairro | | varchar(50) | 1 e 50 | NOT NULL |  |  | Bairro do Cliente |
| cep | | varchar(8) | 8 e 8 | NOT NULL |  |  | CEP do Cliente |
| cidade | | varchar(50) | 1 e 50 | NOT NULL |  |  | Cidade do Cliente |
| uf | | char(2) | 2 e 2 | NOT NULL |  |  | Unidade Federativa (Estado) do Cliente |
| **Índice** | | | | | | | |
| **Nome do índice** | | **Clustered** | **NonClustered** | **Unique** | **Colunas** | | |
| idx\_endereco | |  | X |  | cliente(id\_endereco) INCLUDE(cd\_cliente, nm\_cliente) | | |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **Tabela** | Veículos | | | | | | |
| **Descrição** | Tabela responsável por armazenar os veículos presentes na Locadora | | | | | | |
| **Atributos** | | | | | | | |
| **Nome da Coluna** | | **Tipo do Dado** | **Valor min e max** | **Nulidade** | **PK** | **FK** | **Descrição** |
| placa | | char(7) | 7 e 7 | NOT NULL | X |  | Placa do veículo |
| ano | | date | 4 e 4 | NOT NULL |  |  | Ano do veículo |
| id\_cor | | int | 1 - 2147483647 | NOT NULL |  | X | Código da cor |
| id\_modelo | | int | 1 - 2147483647 | NOT NULL |  | X | Código do modelo |
| **Índice** | | | | | | | |
| **Nome do índice** | | **Clustered** | **NonClustered** | **Unique** | **Colunas** | | |
| idx\_veiculos | | X |  |  | idx\_veiculos | | |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **Tabela** | Aluguel | | | | | | |
| **Descrição** | Tabela responsável pelo armazenamento dos dados de alugueis | | | | | | |
| **Atributos** | | | | | | | |
| **Nome da Coluna** | | **Tipo do Dado** | **Valor min e max** | **Nulidade** | **PK** | **FK** | **Descrição** |
| id\_aluguel | | int | 1 - 2147483647 | NOT NULL | X |  | Código do aluguel |
| cd\_cliente | | int | 1 - 2147483647 | NOT NULL |  | X | Referência da tabela cliente |
| placa | | char(7) | 7 e 7 | NOT NULL |  | X | Referência da tabela veículos |
| data\_inicio | | datetime | 8 e 8 | NOT NULL |  |  | Data de início do aluguel |
| data\_fim | | datetime | 8 e 8 | NOT NULL |  |  | Data de fim do aluguel |
| valor | | decimal(11,2) | 1 - 999999999,99 | NOT NULL |  |  | Valor do aluguel |
| id\_pagamento | | int | 1 - 2147483647 | NOT NULL |  | X | Referência da tabela forma de pagamento |
| **Índice** | | | | | | | |
| **Nome do índice** | | **Clustered** | **NonClustered** | **Unique** | **Colunas** | | |
| idx\_aluguel | | X |  |  | aluguel(id\_aluguel) | | |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **Tabela** | Forma de pagamento | | | | | | |
| **Descrição** | Tabela responsável por armazenar os tipos de pagamentos | | | | | | |
| **Atributos** | | | | | | | |
| **Nome da Coluna** | | **Tipo do Dado** | **Valor min e max** | **Nulidade** | **PK** | **FK** | **Descrição** |
| cd\_forma\_pagamento | | int | 1 - 2147483647 | NOT NULL | X |  | Código da forma de pagamento |
| tipo\_pagamento | | varchar(50) | 1 e 50 | NOT NULL |  |  | Tipo de pagamento |
| **Índice** | | | | | | | |
| **Nome do índice** | | **Clustered** | **NonClustered** | **Unique** | **Colunas** | | |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **Tabela** | Pagamento | | | | | | |
| **Descrição** | Tabela responsável por armazenar os dados dos pagamentos | | | | | | |
| **Atributos** | | | | | | | |
| **Nome da Coluna** | | **Tipo do Dado** | **Valor min e max** | **Nulidade** | **PK** | **FK** | **Descrição** |
| id\_pagamento | | int | 1 - 2147483647 | NOT NULL | X |  | Código do Pagamento |
| cd\_forma\_pgamento | | int | 1 - 2147483647 | NOT NULL |  | X | Referência da tabela Forma de Pagamento |
| valor | | decimal(11,2) | 1 - 999999999,99 | NOT NULL |  |  | Valor do pagamento |
| **Índice** | | | | | | | |
| **Nome do índice** | | **Clustered** | **NonClustered** | **Unique** | **Colunas** | | |

**Script dos comandos DDL para criação do Banco de dados**

CREATE TABLE marca (

id\_marca INT identity PRIMARY KEY,

marca VARCHAR(20)

);

CREATE TABLE modelo (

id\_modelo INT identity PRIMARY KEY,

modelo VARCHAR(20),

id\_marca INT,

FOREIGN KEY (id\_marca) REFERENCES marca(id\_marca)

);

CREATE TABLE cor (

id\_cor INT identity PRIMARY KEY,

cor VARCHAR(20)

);

CREATE TABLE endereco (

id\_endereco INT identity PRIMARY KEY,

uf CHAR(2),

cep CHAR(15),

cidade VARCHAR(100),

bairro VARCHAR(50),

rua VARCHAR(200),

numero INT

);

CREATE TABLE cliente (

cd\_cliente INT identity PRIMARY KEY,

nm\_cliente VARCHAR(200),

telefone BIGINT,

email VARCHAR(200),

cpf BIGINT,

id\_endereco INT,

FOREIGN KEY (id\_endereco) REFERENCES endereco(id\_endereco)

);

CREATE TABLE veiculos (

placa CHAR(7) PRIMARY KEY,

ano DATE,

id\_cor INT,

id\_modelo INT,

FOREIGN KEY (id\_cor) REFERENCES cor(id\_cor),

FOREIGN KEY (id\_modelo) REFERENCES modelo(id\_modelo)

);

CREATE TABLE forma\_pagamento (

cd\_forma\_pagamento INT identity PRIMARY KEY,

tipo\_pagamento VARCHAR(50)

);

CREATE TABLE pagamento (

id\_pagamento INT identity PRIMARY KEY,

cd\_forma\_pagamento INT,

valor DECIMAL(11,2),

FOREIGN KEY (cd\_forma\_pagamento) REFERENCES forma\_pagamento(cd\_forma\_pagamento)

);

CREATE TABLE aluguel (

id\_aluguel INT identity PRIMARY KEY,

cd\_cliente INT,

placa CHAR(7),

id\_pagamento INT,

data\_inicio DATETIME,

data\_fim DATETIME,

valor DECIMAL(11,2),

FOREIGN KEY (cd\_cliente) REFERENCES cliente(cd\_cliente),

FOREIGN KEY (placa) REFERENCES veiculos(placa),

FOREIGN KEY (id\_pagamento) REFERENCES pagamento(id\_pagamento)

);

**Script que popula as tabelas do Banco de dados**

-- Insert statements for table marca

INSERT INTO marca (marca) VALUES ('Ford');

INSERT INTO marca (marca) VALUES ('Toyota');

INSERT INTO marca (marca) VALUES ('Honda');

INSERT INTO marca (marca) VALUES ('Chevrolet');

INSERT INTO marca (marca) VALUES ('Nissan');

INSERT INTO marca (marca) VALUES ('Volkswagen');

INSERT INTO marca (marca) VALUES ('Hyundai');

INSERT INTO marca (marca) VALUES ('Kia');

INSERT INTO marca (marca) VALUES ('BMW');

INSERT INTO marca (marca) VALUES ('Mercedes-Benz');

-- Insert statements for table modelo

INSERT INTO modelo (modelo, id\_marca) VALUES ('Fiesta', 1);

INSERT INTO modelo (modelo, id\_marca) VALUES ('Corolla', 2);

INSERT INTO modelo (modelo, id\_marca) VALUES ('Civic', 3);

INSERT INTO modelo (modelo, id\_marca) VALUES ('Cruze', 4);

INSERT INTO modelo (modelo, id\_marca) VALUES ('Sentra', 5);

INSERT INTO modelo (modelo, id\_marca) VALUES ('Golf', 6);

INSERT INTO modelo (modelo, id\_marca) VALUES ('Elantra', 7);

INSERT INTO modelo (modelo, id\_marca) VALUES ('Optima', 8);

INSERT INTO modelo (modelo, id\_marca) VALUES ('3 Series', 9);

INSERT INTO modelo (modelo, id\_marca) VALUES ('C-Class', 10);

-- Insert statements for table cor

INSERT INTO cor (cor) VALUES ('Vermelho');

INSERT INTO cor (cor) VALUES ('Azul');

INSERT INTO cor (cor) VALUES ('Preto');

INSERT INTO cor (cor) VALUES ('Branco');

INSERT INTO cor (cor) VALUES ('Prata');

INSERT INTO cor (cor) VALUES ('Cinza');

INSERT INTO cor (cor) VALUES ('Verde');

INSERT INTO cor (cor) VALUES ('Amarelo');

INSERT INTO cor (cor) VALUES ('Marrom');

INSERT INTO cor (cor) VALUES ('Roxo');

-- Insert statements for table endereco

INSERT INTO endereco (uf, cep, cidade, bairro, rua, numero) VALUES ('SC', '88800-000', 'Criciuma', 'Centro', 'Rua A', 1);

INSERT INTO endereco (uf, cep, cidade, bairro, rua, numero) VALUES ('SC', '88701-000', 'Tubarao', 'Vila Moema', 'Rua B', 2);

INSERT INTO endereco (uf, cep, cidade, bairro, rua, numero) VALUES ('SC', '88790-000', 'Laguna', 'Centro', 'Rua C', 3);

INSERT INTO endereco (uf, cep, cidade, bairro, rua, numero) VALUES ('SC', '88820-000', 'Icara', 'Presidente Vargas', 'Rua D', 4);

INSERT INTO endereco (uf, cep, cidade, bairro, rua, numero) VALUES ('SC', '88900-000', 'Ararangua', 'Cidade Alta', 'Rua E', 5);

INSERT INTO endereco (uf, cep, cidade, bairro, rua, numero) VALUES ('SC', '88801-000', 'Criciuma', 'Comerciario', 'Rua F', 6);

INSERT INTO endereco (uf, cep, cidade, bairro, rua, numero) VALUES ('SC', '88702-000', 'Tubarao', 'Oficinas', 'Rua G', 7);

INSERT INTO endereco (uf, cep, cidade, bairro, rua, numero) VALUES ('SC', '88791-000', 'Laguna', 'Progresso', 'Rua H', 8);

INSERT INTO endereco (uf, cep, cidade, bairro, rua, numero) VALUES ('SC', '88821-000', 'Icara', 'Centro', 'Rua I', 9);

INSERT INTO endereco (uf, cep, cidade, bairro, rua, numero) VALUES ('SC', '88901-000', 'Ararangua', 'Centro', 'Rua J', 10);

-- Insert statements for table cliente

INSERT INTO cliente (nm\_cliente, telefone, email, cpf, id\_endereco) VALUES ('Joao Silva', 11987654321, 'joao.silva@example.com', 12345678901, 1);

INSERT INTO cliente (nm\_cliente, telefone, email, cpf, id\_endereco) VALUES ('Maria Santos', 21987654321, 'maria.santos@example.com', 12345678902, 2);

INSERT INTO cliente (nm\_cliente, telefone, email, cpf, id\_endereco) VALUES ('Pedro Costa', 31987654321, 'pedro.costa@example.com', 12345678903, 3);

INSERT INTO cliente (nm\_cliente, telefone, email, cpf, id\_endereco) VALUES ('Ana Lima', 41987654321, 'ana.lima@example.com', 12345678904, 4);

INSERT INTO cliente (nm\_cliente, telefone, email, cpf, id\_endereco) VALUES ('Carlos Souza', 51987654321, 'carlos.souza@example.com', 12345678905, 5);

INSERT INTO cliente (nm\_cliente, telefone, email, cpf, id\_endereco) VALUES ('Fernanda Pereira', 61987654321, 'fernanda.pereira@example.com', 12345678906, 6);

INSERT INTO cliente (nm\_cliente, telefone, email, cpf, id\_endereco) VALUES ('Lucas Oliveira', 71987654321, 'lucas.oliveira@example.com', 12345678907, 7);

INSERT INTO cliente (nm\_cliente, telefone, email, cpf, id\_endereco) VALUES ('Juliana Almeida', 81987654321, 'juliana.almeida@example.com', 12345678908, 8);

INSERT INTO cliente (nm\_cliente, telefone, email, cpf, id\_endereco) VALUES ('Rafael Fernandes', 91987654321, 'rafael.fernandes@example.com', 12345678909, 9);

INSERT INTO cliente (nm\_cliente, telefone, email, cpf, id\_endereco) VALUES ('Patricia Gomes', 101987654321, 'patricia.gomes@example.com', 12345678910, 10);

-- Insert statements for table veiculos

INSERT INTO veiculos (placa, ano, id\_cor, id\_modelo) VALUES ('ABC1234', '2015-01-01', 1, 1);

INSERT INTO veiculos (placa, ano, id\_cor, id\_modelo) VALUES ('DEF5678', '2016-01-01', 2, 2);

INSERT INTO veiculos (placa, ano, id\_cor, id\_modelo) VALUES ('GHI9012', '2017-01-01', 3, 3);

INSERT INTO veiculos (placa, ano, id\_cor, id\_modelo) VALUES ('JKL3456', '2018-01-01', 4, 4);

INSERT INTO veiculos (placa, ano, id\_cor, id\_modelo) VALUES ('MNO7890', '2019-01-01', 5, 5);

INSERT INTO veiculos (placa, ano, id\_cor, id\_modelo) VALUES ('PQR2345', '2020-01-01', 6, 6);

INSERT INTO veiculos (placa, ano, id\_cor, id\_modelo) VALUES ('STU6789', '2021-01-01', 7, 7);

INSERT INTO veiculos (placa, ano, id\_cor, id\_modelo) VALUES ('VWX1234', '2022-01-01', 8, 8);

INSERT INTO veiculos (placa, ano, id\_cor, id\_modelo) VALUES ('YZA5678', '2023-01-01', 9, 9);

INSERT INTO veiculos (placa, ano, id\_cor, id\_modelo) VALUES ('BCD9012', '2024-01-01', 10, 10);

-- Insert statements for table forma\_pagamento

INSERT INTO forma\_pagamento (tipo\_pagamento) VALUES ('Cartão de Crédito');

INSERT INTO forma\_pagamento (tipo\_pagamento) VALUES ('Cartão de Débito');

INSERT INTO forma\_pagamento (tipo\_pagamento) VALUES ('Dinheiro');

INSERT INTO forma\_pagamento (tipo\_pagamento) VALUES ('Pix');

INSERT INTO forma\_pagamento (tipo\_pagamento) VALUES ('PayPal');

INSERT INTO forma\_pagamento (tipo\_pagamento) VALUES ('Apple Pay');

INSERT INTO forma\_pagamento (tipo\_pagamento) VALUES ('Google Pay');

INSERT INTO forma\_pagamento (tipo\_pagamento) VALUES ('Cheque');

INSERT INTO forma\_pagamento (tipo\_pagamento) VALUES ('Transf. Bancária');

INSERT INTO forma\_pagamento (tipo\_pagamento) VALUES ('Voucher');

-- Insert statements for table pagamento

INSERT INTO pagamento (cd\_forma\_pagamento, valor) VALUES (1, 150.50);

INSERT INTO pagamento (cd\_forma\_pagamento, valor) VALUES (2, 200.75);

INSERT INTO pagamento (cd\_forma\_pagamento, valor) VALUES (3, 100.00);

INSERT INTO pagamento (cd\_forma\_pagamento, valor) VALUES (4, 300.20);

INSERT INTO pagamento (cd\_forma\_pagamento, valor) VALUES (5, 250.00);

INSERT INTO pagamento (cd\_forma\_pagamento, valor) VALUES (6, 175.80);

INSERT INTO pagamento (cd\_forma\_pagamento, valor) VALUES (7, 225.90);

INSERT INTO pagamento (cd\_forma\_pagamento, valor) VALUES (8, 400.60);

INSERT INTO pagamento (cd\_forma\_pagamento, valor) VALUES (9, 320.40);

INSERT INTO pagamento (cd\_forma\_pagamento, valor) VALUES (10, 500.00);

-- Insert statements for table aluguel

INSERT INTO aluguel (cd\_cliente, placa, id\_pagamento, data\_inicio, data\_fim, valor) VALUES (1, 'ABC1234', 1, '2023-01-01 09:00:00', '2023-01-05 09:00:00', 150.50);

INSERT INTO aluguel (cd\_cliente, placa, id\_pagamento, data\_inicio, data\_fim, valor) VALUES (2, 'DEF5678', 2, '2023-02-01 09:00:00', '2023-02-07 09:00:00', 200.75);

INSERT INTO aluguel (cd\_cliente, placa, id\_pagamento, data\_inicio, data\_fim, valor) VALUES (3, 'GHI9012', 3, '2023-03-01 09:00:00', '2023-03-03 09:00:00', 100.00);

INSERT INTO aluguel (cd\_cliente, placa, id\_pagamento, data\_inicio, data\_fim, valor) VALUES (4, 'JKL3456', 4, '2023-04-01 09:00:00', '2023-04-10 09:00:00', 300.20);

INSERT INTO aluguel (cd\_cliente, placa, id\_pagamento, data\_inicio, data\_fim, valor) VALUES (5, 'MNO7890', 5, '2023-05-01 09:00:00', '2023-05-08 09:00:00', 250.00);

INSERT INTO aluguel (cd\_cliente, placa, id\_pagamento, data\_inicio, data\_fim, valor) VALUES (6, 'PQR2345', 6, '2023-06-01 09:00:00', '2023-06-05 09:00:00', 175.80);

INSERT INTO aluguel (cd\_cliente, placa, id\_pagamento, data\_inicio, data\_fim, valor) VALUES (7, 'STU6789', 7, '2023-07-01 09:00:00', '2023-07-10 09:00:00', 225.90);

INSERT INTO aluguel (cd\_cliente, placa, id\_pagamento, data\_inicio, data\_fim, valor) VALUES (8, 'VWX1234', 8, '2023-08-01 09:00:00', '2023-08-12 09:00:00', 400.60);

INSERT INTO aluguel (cd\_cliente, placa, id\_pagamento, data\_inicio, data\_fim, valor) VALUES (9, 'YZA5678', 9, '2023-09-01 09:00:00', '2023-09-05 09:00:00', 320.40);

INSERT INTO aluguel (cd\_cliente, placa, id\_pagamento, data\_inicio, data\_fim, valor) VALUES (10, 'BCD9012', 10, '2023-10-01 09:00:00', '2023-10-15 09:00:00', 500.00);

**TRIGGER**

-- Criar a nova trigger DescontoPIX

CREATE TRIGGER DescontoPIX

ON pagamento

AFTER INSERT

AS

BEGIN

SET NOCOUNT ON;

DECLARE @id\_pagamento INT;

DECLARE @valor DECIMAL(11,2);

DECLARE @cd\_forma\_pagamento INT;

SELECT

@id\_pagamento = id\_pagamento,

@valor = valor,

@cd\_forma\_pagamento = cd\_forma\_pagamento

FROM inserted;

-- Verificar se a forma de pagamento é PIX

IF EXISTS (SELECT 1 FROM forma\_pagamento WHERE cd\_forma\_pagamento = @cd\_forma\_pagamento AND tipo\_pagamento = 'Pix')

BEGIN

-- Aplicar desconto de 10%

UPDATE pagamento

SET valor = @valor \* 0.9

WHERE id\_pagamento = @id\_pagamento;

END;

END;

GO

-- Inserir um pagamento via PIX

INSERT INTO pagamento (cd\_forma\_pagamento, valor)

VALUES (7, 1500.00); -- Valor antes do desconto: 1500.00 (7 pode ser diferente dependendo de qual o id do Pix)

-- Verificar se o desconto foi aplicado

SELECT \* FROM pagamento;

**INDEX**

CREATE NONCLUSTERED INDEX idx\_endereco\_rua ON endereco(rua) INCLUDE(id\_endereco, cidade, bairro);

CREATE NONCLUSTERED INDEX idx\_cliente\_endereco ON cliente(id\_endereco) INCLUDE(cd\_cliente, nm\_cliente);

CREATE NONCLUSTERED INDEX idx\_cliente\_telefone ON cliente(telefone) INCLUDE(cd\_cliente, email, cpf);

CREATE NONCLUSTERED INDEX idx\_veiculos\_modelo ON veiculos(id\_modelo) INCLUDE(placa, id\_cor);

CREATE NONCLUSTERED INDEX idx\_aluguel\_cliente ON aluguel(cd\_cliente) INCLUDE(id\_aluguel, placa);

CREATE NONCLUSTERED INDEX idx\_aluguel\_pagamento ON aluguel(id\_pagamento) INCLUDE(id\_aluguel, valor);

-- Exemplo de uso dos índices:

-- Pesquisa sem índice nonclustered

-- SELECT data\_inicio, data\_fim, valor FROM aluguel

-- WHERE valor = 400.60;

-- Pesquisa com índice nonclustered e include

-- SELECT nm\_cliente, id\_endereco FROM cliente

-- WHERE id\_endereco = 3;

**Principais consultas mapeadas baseadas em regras de negócio (mínimo 4)**

**1 - Listar todos os clientes que alugaram veículos de uma determinada cor em um período específico:**

CREATE FUNCTION ListarClientesPorCorEPeriodo(@cor VARCHAR(20), @data\_inicio DATETIME, @data\_fim DATETIME)

RETURNS TABLE

AS

RETURN

SELECT

c.nm\_cliente,

c.telefone,

c.email,

c.cpf

FROM

cliente c

JOIN

aluguel a ON c.cd\_cliente = a.cd\_cliente

JOIN

veiculos v ON a.placa = v.placa

JOIN

cor co ON v.id\_cor = co.id\_cor

WHERE

co.cor = @cor

AND a.data\_inicio BETWEEN @data\_inicio AND @data\_fim

GROUP BY

c.nm\_cliente,

c.telefone,

c.email,

c.cpf;

-- Exemplo de uso da função:

-- SELECT \*

-- FROM ListarClientesPorCorEPeriodo('Preto', '2023-01-01', '2023-12-31');

**2 - Quais são os modelos de veículos mais alugados em cada estado e qual é o valor total gerado por esses modelos em cada estado, depois gere um ranking dos modelos mais alugados e seus respectivos estados.**

-- Subconsulta para obter a contagem de aluguéis e valor total gerado por modelo e estado

WITH alugueis\_por\_modelo\_estado AS (

SELECT

m.modelo,

e.uf,

COUNT(a.id\_aluguel) AS total\_alugueis,

SUM(a.valor) AS valor\_total

FROM

aluguel a

JOIN veiculos v ON a.placa = v.placa

JOIN modelo m ON v.id\_modelo = m.id\_modelo

JOIN cliente c ON a.cd\_cliente = c.cd\_cliente

JOIN endereco e ON c.id\_endereco = e.id\_endereco

GROUP BY

m.modelo, e.uf

),

-- Subconsulta para obter o modelo mais alugado em cada estado

modelo\_mais\_alugado\_por\_estado AS (

SELECT

uf,

modelo,

total\_alugueis,

valor\_total,

RANK() OVER (PARTITION BY uf ORDER BY total\_alugueis DESC, valor\_total DESC) AS ranking

FROM

alugueis\_por\_modelo\_estado

)

-- Selecionar os modelos mais alugados em cada estado

SELECT

uf,

modelo,

total\_alugueis,

valor\_total

FROM

modelo\_mais\_alugado\_por\_estado

WHERE

ranking = 1

ORDER BY

uf;

-- Ranking dos modelos mais alugados e seus respectivos estados

SELECT

modelo,

uf,

total\_alugueis,

valor\_total,

RANK() OVER (ORDER BY total\_alugueis DESC, valor\_total DESC) AS ranking

FROM

alugueis\_por\_modelo\_estado

ORDER BY

ranking;

**3 - Quantos aluguéis foram feitos em 2023 para um modelo específico de carro, em Santa Catarina (SC), e pagos somente via cheque.**

SELECT

m.modelo,

COUNT(a.id\_aluguel) AS total\_alugueis

FROM

aluguel a

JOIN veiculos v ON a.placa = v.placa

JOIN modelo m ON v.id\_modelo = m.id\_modelo

JOIN cliente c ON a.cd\_cliente = c.cd\_cliente

JOIN endereco e ON c.id\_endereco = e.id\_endereco

JOIN pagamento p ON a.id\_pagamento = p.id\_pagamento

JOIN forma\_pagamento fp ON p.cd\_forma\_pagamento = fp.cd\_forma\_pagamento

WHERE

e.uf = 'SC'

AND fp.tipo\_pagamento = 'Cheque'

AND YEAR(a.data\_inicio) = 2023

GROUP BY

m.modelo

ORDER BY

total\_alugueis DESC;

**4 - Listar todos os clientes que alugaram veículos de uma determinada marca e que também fizeram pagamentos usando crédito, incluindo a última data de aluguel de cada cliente.**

CREATE OR ALTER PROCEDURE ListarClientesPorMarcaEFormaPagamento

@p\_marca\_nome VARCHAR(20),

@p\_tipo\_pagamento VARCHAR(50)

AS

BEGIN

SELECT

c.nm\_cliente,

c.telefone,

c.email,

c.cpf,

MAX(a.data\_fim) AS ultima\_data\_aluguel

FROM

cliente c

JOIN

aluguel a ON c.cd\_cliente = a.cd\_cliente

JOIN

veiculos v ON a.placa = v.placa

JOIN

modelo m ON v.id\_modelo = m.id\_modelo

JOIN

marca ma ON m.id\_marca = ma.id\_marca

JOIN

pagamento p ON a.id\_pagamento = p.id\_pagamento

JOIN

forma\_pagamento fp ON p.cd\_forma\_pagamento = fp.cd\_forma\_pagamento

WHERE

ma.marca = @p\_marca\_nome AND

fp.tipo\_pagamento = @p\_tipo\_pagamento

GROUP BY

c.cd\_cliente, c.nm\_cliente, c.telefone, c.email, c.cpf;

END;

EXEC ListarClientesPorMarcaEFormaPagamento @p\_marca\_nome = 'Ford', @p\_tipo\_pagamento = 'Cartão de Crédito';

--Valor da marca e do tipo de pagamento pode mudar.