

# Projeto Final Banco de Dados II

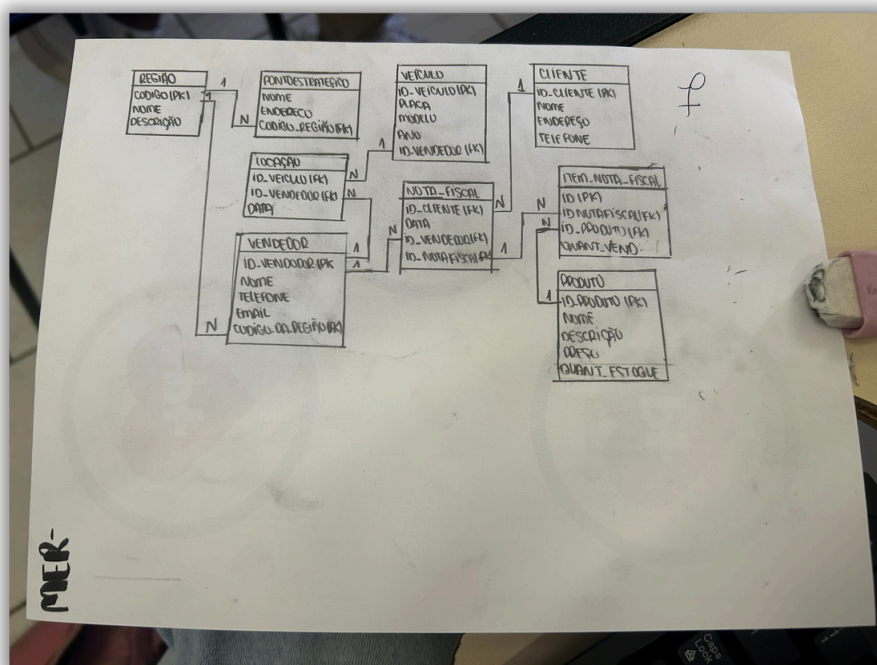
Giovana Marsigli Rodrigues - 3DS AMS

**Indústria de Beleza**

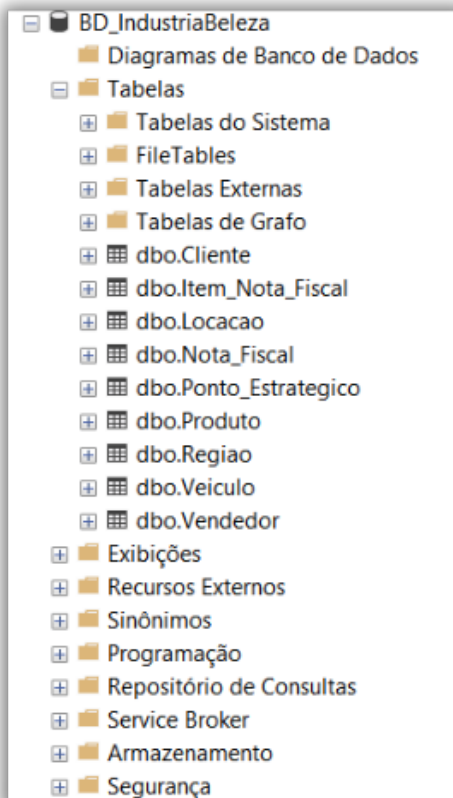
**ESTUDO DO CASO:** O Departamento de Vendas da Indústria Beleza Ltda, após estudos de mercado, verificou que para atingir seus objetivos seria necessário adquirir frota de veículos próprios para motorizar seus vendedores. O mercado consumidor foi dividido em regiões de venda; foram estabelecidos percursos de entrega abrangendo pontos estratégicos dessas regiões e vendedores foram designados para cobrir estes percursos. Um sistema deve ser construído para administração da nova sistemática de vendas adotada pela empresa. Após entrevistas com o gerente da área, foram obtidas as seguintes informações:

- Cada região é identificada por um código;
- Uma região é composta de vários pontos estratégicos;
- As regiões não têm pontos estratégicos em comum;
- O vendedor tem a responsabilidade de cobrir uma região;
- Uma região pode ser coberta por vários vendedores;
- A cada dia, um veículo fica sob a responsabilidade de um vendedor;
- Um vendedor pode vender quaisquer itens ativos da tabela de produtos;
- O vendedor é responsável pela identificação de cada cliente consumidor na nota fiscal;
- A nota fiscal contendo identificação do vendedor, itens e quantidades vendidas é exigida para comprovação da venda.

**a) Crie o MER (Modelo Entidade Relacionamento) no papel.**



## b) Crie esta base de dados no SSMS.



## c) Criação das tabelas

```
-- Tabela Regiao
CREATE TABLE Regiao (
    codigo INT PRIMARY KEY,
    nome VARCHAR(100),
    descricao TEXT
);

-- Tabela Ponto_Estrategico
CREATE TABLE Ponto_Estrategico (
    id INT PRIMARY KEY,
    nome VARCHAR(100),
    endereco TEXT,
    codigo_regiao INT,
    FOREIGN KEY (codigo_regiao) REFERENCES Regiao(codigo)
);

-- Tabela Cliente
CREATE TABLE Cliente (
    id_cliente INT PRIMARY KEY,
    nome VARCHAR(100),
    endereco TEXT,
    telefone VARCHAR(15)
);
```

### c) Criação das tabelas

```
-- Tabela Vendedor
CREATE TABLE Vendedor (
    id_vendedor INT PRIMARY KEY,
    nome VARCHAR(100),
    telefone VARCHAR(15),
    email VARCHAR(100),
    codigo_da_regiao INT,
    FOREIGN KEY (codigo_da_regiao) REFERENCES Regiao(codigo)
);

-- Tabela Veiculo
CREATE TABLE Veiculo (
    id_veiculo INT PRIMARY KEY,
    placa VARCHAR(10),
    modelo VARCHAR(50),
    ano INT,
    id_vendedor INT,
    FOREIGN KEY (id_vendedor) REFERENCES Vendedor(id_vendedor)
);

-- Tabela Locacao
CREATE TABLE Locacao (
    id_veiculo INT,
    id_vendedor INT,
    data DATE,
    PRIMARY KEY (id_veiculo, id_vendedor, data),
    FOREIGN KEY (id_veiculo) REFERENCES Veiculo(id_veiculo),
    FOREIGN KEY (id_vendedor) REFERENCES Vendedor(id_vendedor)
);
```

```
-- Tabela Produto
CREATE TABLE Produto (
    id_produto INT PRIMARY KEY,
    nome VARCHAR(100),
    descricao TEXT,
    preco DECIMAL(10, 2),
    quant_estoque INT
);

-- Tabela Nota_Fiscal
CREATE TABLE Nota_Fiscal (
    id_nota_fiscal INT PRIMARY KEY,
    id_cliente INT,
    data DATE,
    id_vendedor INT,
    FOREIGN KEY (id_cliente) REFERENCES Cliente(id_cliente),
    FOREIGN KEY (id_vendedor) REFERENCES Vendedor(id_vendedor)
);

-- Tabela Item_Nota_Fiscal
CREATE TABLE Item_Nota_Fiscal (
    id INT PRIMARY KEY,
    id_nota_fiscal INT,
    id_produto INT,
    quant_vendida INT,
    FOREIGN KEY (id_nota_fiscal) REFERENCES Nota_Fiscal(id_nota_fiscal),
    FOREIGN KEY (id_produto) REFERENCES Produto(id_produto)
);
```

## d) Incluir registros nas tabelas

### Tabela Região

```
-- Inserts para a tabela Regiao
INSERT INTO Regiao (codigo, nome, descricao) VALUES
(1, 'Região Norte', 'Estados do norte do Brasil'),
(2, 'Região Sul', 'Estados do sul do Brasil'),
(3, 'Região Sudeste', 'Estados do sudeste do Brasil'),
(4, 'Região Centro-Oeste', 'Estados do centro-oeste do Brasil'),
(5, 'Região Nordeste', 'Estados do nordeste do Brasil');
SELECT * FROM Regiao;
```

91 %

Resultados Mensagens

	codigo	nome	descricao
1	1	Região Norte	Estados do norte do Brasil
2	2	Região Sul	Estados do sul do Brasil
3	3	Região Sudeste	Estados do sudeste do Brasil
4	4	Região Centro-Oeste	Estados do centro-oeste do Brasil
5	5	Região Nordeste	Estados do nordeste do Brasil

### Tabela Ponto\_Estrategico

```
-- Inserts para a tabela Ponto_Estrategico
INSERT INTO Ponto_Estrategico (id, nome, endereco, codigo_regiao) VALUES
(1, 'Ponto de Vendas Amazonas', 'Av. Manaus, 200, Manaus', 1),
(2, 'Ponto Sul', 'Rua Paraná, 45, Curitiba', 2),
(3, 'Ponto de Acesso São Paulo', 'Rua Paulista, 100, São Paulo', 3),
(4, 'Centro Oeste Atacadista', 'Av. Goiás, 75, Goiânia', 4),
(5, 'Nordeste Center', 'Rua Bahia, 88, Salvador', 5);
SELECT * FROM Ponto_Estrategico;
```

91 %

Resultados Mensagens

	id	nome	endereco	codigo_regiao
1	1	Ponto de Vendas Amazonas	Av. Manaus, 200, Manaus	1
2	2	Ponto Sul	Rua Paraná, 45, Curitiba	2
3	3	Ponto de Acesso São Paulo	Rua Paulista, 100, São Paulo	3
4	4	Centro Oeste Atacadista	Av. Goiás, 75, Goiânia	4
5	5	Nordeste Center	Rua Bahia, 88, Salvador	5

### Tabela Cliente

```
-- Inserts para a tabela Cliente
INSERT INTO Cliente (id_cliente, nome, endereco, telefone) VALUES
(1, 'Odair', 'Rua das Flores, 123, Brasília', '(61) 91234-5678'),
(2, 'Juliana', 'Avenida Brasil, 456, Rio de Janeiro', '(21) 98765-4321'),
(3, 'Giullia', 'Rua dos Limoeiros, 789, Belo Horizonte', '(31) 99876-5432'),
(4, 'Larissa', 'Praça das Águas, 321, São Paulo', '(11) 93456-7890'),
(5, 'Alexandre', 'Alameda das Palmeiras, 654, Porto Alegre', '(51) 91234-0987');
SELECT * FROM Cliente;
```

91 %

Resultados Mensagens

	id_cliente	nome	endereco	telefone
1	1	Odair	Rua das Flores, 123, Brasília	(61) 91234-5678
2	2	Juliana	Avenida Brasil, 456, Rio de Janeiro	(21) 98765-4321
3	3	Giullia	Rua dos Limoeiros, 789, Belo Horizonte	(31) 99876-5432
4	4	Larissa	Praça das Águas, 321, São Paulo	(11) 93456-7890
5	5	Alexandre	Alameda das Palmeiras, 654, Porto Alegre	(51) 91234-0987

## Tabela Vendedor

```
-- Inserts para a tabela Vendedor
INSERT INTO Vendedor (id_vendedor, nome, telefone, email, codigo_da_regiao) VALUES
(1, 'Mariana', '(61) 90000-0001', 'mariana@gmail.com', 1),
(2, 'Laura', '(21) 90000-0002', 'laura@gmail.com', 2),
(3, 'Luiza', '(31) 90000-0003', 'luiza@gmail.com', 3),
(4, 'Lucia', '(11) 90000-0004', 'lucia@gmail.com', 4),
(5, 'João', '(51) 90000-0005', 'joao@gmail.com', 5);
SELECT * FROM Vendedor;
```

id_vendedor	nome	telefone	email	codigo_da_regiao
1	Mariana	(61) 90000-0001	mariana@gmail.com	1
2	Laura	(21) 90000-0002	laura@gmail.com	2
3	Luiza	(31) 90000-0003	luiza@gmail.com	3
4	Lucia	(11) 90000-0004	lucia@gmail.com	4
5	João	(51) 90000-0005	joao@gmail.com	5

## Tabela Veiculo

```
-- Inserts para a tabela Veiculo
INSERT INTO Veiculo (id_veiculo, placa, modelo, ano, id_vendedor) VALUES
(1, 'ABC-1A2', 'Fusca', 1974, 1),
(2, 'XYZ-3B4', 'Civic', 2020, 2),
(3, 'DEF-5C6', 'Corolla', 2021, 3),
(4, 'GHI-7D8', 'Fiesta', 2022, 4),
(5, 'JKL-9E0', 'Onix', 2023, 5);
SELECT * FROM Veiculo;
```

id_veiculo	placa	modelo	ano	id_vendedor
1	ABC-1A2	Fusca	1974	1
2	XYZ-3B4	Civic	2020	2
3	DEF-5C6	Corolla	2021	3
4	GHI-7D8	Fiesta	2022	4
5	JKL-9E0	Onix	2023	5

## Tabela Locacao

```
-- Inserts para a tabela Locacao
INSERT INTO Locacao (id_veiculo, id_vendedor, data) VALUES
(1, 1, '2023-10-01'),
(2, 2, '2023-10-05'),
(3, 3, '2023-10-10'),
(4, 4, '2023-10-15'),
(5, 5, '2023-10-20');
SELECT * FROM Locacao;
```

id_veiculo	id_vendedor	data
1	1	2023-10-01
2	2	2023-10-05
3	3	2023-10-10
4	4	2023-10-15
5	5	2023-10-20

## Tabela Produto

```
-- Inserts para a tabela Produto
INSERT INTO Produto (id_produto, nome, descricao, preco, quant_estoque) VALUES
(1, 'Rádio', 'Rádio para carro', 12.99, 150),
(2, 'Capas para banco', 'Protetores para os bancos do carro', 5.50, 200),
(3, 'Câmeras de ré', 'Câmeras que ajudam na visualização ao dar marcha à ré.', 3.00, 300),
(4, 'Suportes para celular', 'Permite fixar o celular em um local acessível.', 1.50, 1000),
(5, 'Organizadores', 'Organizadores para carro', 4.75, 250),
(6, 'Antena', 'Antena', 3.50, 100);
SELECT * FROM Produto;
```

id_produto	nome	descricao	preco	quant_estoque
1	Rádio	Rádio para carro	12.99	150
2	Capas para banco	Protetores para os bancos do carro	5.50	200
3	Câmeras de ré	Câmeras que ajudam na visualização ao dar marcha...	3.00	300
4	Suportes para celular	Permite fixar o celular em um local acessível.	1.50	1000
5	Organizadores	Organizadores para carro	4.75	250
6	Antena	Antena	3.50	100

## Tabela Nota\_Fiscal

```
-- Inserts para a tabela Nota_Fiscal
INSERT INTO Nota_Fiscal (id_nota_fiscal, id_cliente, data, id_vendedor) VALUES
(1, 1, '2023-10-02', 1),
(2, 2, '2023-10-06', 2),
(3, 3, '2023-10-11', 3),
(4, 4, '2023-10-16', 4),
(5, 5, '2023-10-21', 5);
SELECT * FROM Nota_Fiscal;
```

	id_nota_fiscal	id_cliente	data	id_vendedor
1	1	1	2023-10-02	1
2	2	2	2023-10-06	2
3	3	3	2023-10-11	3
4	4	4	2023-10-16	4
5	5	5	2023-10-21	5

## Tabela Item\_Nota\_Fiscal

```
-- Inserts para a tabela Item_Nota_Fiscal
INSERT INTO Item_Nota_Fiscal (id, id_nota_fiscal, id_produto, quant_vendida) VALUES
(1, 1, 1, 3),
(2, 2, 2, 1),
(3, 3, 3, 5),
(4, 4, 4, 10),
(5, 5, 5, 2);
SELECT * FROM Item_Nota_Fiscal;
```

	id	id_nota_fiscal	id_produto	quant_vendida
1	1	1	1	3
2	2	2	2	1
3	3	3	3	5
4	4	4	4	10
5	5	5	5	2

### e) Resolva os exercícios

- a. Listar todos os pontos estratégicos de cada região.

```
--A - Listar todos os pontos estratégicos de cada região.
SELECT nome AS Nome_Ponto_Estrategico, (SELECT nome FROM Regiao WHERE Regiao.codigo = Ponto_Estrategico.codigo_regiao) AS Nome_Regiao
FROM Ponto_Estrategico;
```

	Nome_Ponto_Estrategico	Nome_Regiao
1	Ponto de Vendas Amazonas	Região Norte
2	Ponto Sul	Região Sul
3	Ponto de Acesso São Paulo	Região Sudeste
4	Centro Oeste Atacadista	Região Centro-Oeste
5	Nordeste Center	Região Nordeste

- b. Listar os nomes das regiões cadastradas.

```
--B - Listar os nomes das regiões cadastradas.
SELECT nome AS Nome_Regiao
FROM Regiao;
```

	Nome_Regiao
1	Região Norte
2	Região Sul
3	Região Sudeste
4	Região Centro-Oeste
5	Região Nordeste



- c. Listar todos os vendedores e quais veículos que eles utilizaram no último mês.

```
--C - Listar todos os vendedores e quais veículos que eles utilizaram no último mês.
SELECT V.nome AS Nome_Vendedor, VE.modelo AS Modelo_Veiculo, L.data AS Data_Utilizacao
FROM Vendedor V
JOIN Veiculo VE ON V.id_vendedor = VE.id_vendedor
```

	Nome_Regiao
1	Região Norte
2	Região Sul
3	Região Sudeste
4	Região Centro-Oeste
5	Região Nordeste

- d. Listar todos os vendedores responsáveis por cada região.

```
--D - Listar todos os vendedores responsáveis por cada região.
SELECT (SELECT Regiao.nome FROM Regiao WHERE Regiao.codigo = Vendedor.codigo_da_regiao) AS Nome_Regiao, Vendedor.nome AS Nome_Vendedor
FROM Vendedor;
```

	Nome_Regiao	Nome_Vendedor
1	Região Norte	Mariana
2	Região Sul	Laura
3	Região Sudeste	Luiza
4	Região Centro-Oeste	Lucia
5	Região Nordeste	João

- e. Todos os produtos vendidos por um determinado <vendedor>.

```
--E - Todos os produtos vendidos por um determinado <vendedor>.
DECLARE @id_vendedor INT = 2; -- ID de Laura
SELECT V.nome AS Nome_Vendedor, P.nome AS Nome_Produto
FROM Produto P
JOIN Item_Nota_Fiscal INF ON P.id_produto = INF.id_produto
JOIN Nota_Fiscal NF ON INF.id_nota_fiscal = NF.id_nota_fiscal
JOIN Vendedor V ON NF.id_vendedor = V.id_vendedor
WHERE NF.id_vendedor = @id_vendedor;
```

	Nome_Vendedor	Nome_Produto
1	Laura	Capas para banco

- f. Todos os vendedores que venderam um determinado .

```
--F - Todos os vendedores que venderam um determinado <produto>.
DECLARE @id_produto INT = 1;
SELECT P.nome AS Nome_Produto, V.nome AS Nome_Vendedor
FROM Vendedor V
JOIN Nota_Fiscal NF ON V.id_vendedor = NF.id_vendedor
JOIN Item_Nota_Fiscal INF ON NF.id_nota_fiscal = INF.id_nota_fiscal
JOIN Produto P ON INF.id_produto = P.id_produto
WHERE INF.id_produto = @id_produto;
```

	Nome_Produto	Nome_Vendedor
1	Rádio	Mariana

- g. Todos os produtos que ainda não foram vendidos.

```
--G - Todos os produtos que ainda não foram vendidos.
SELECT Produto.nome AS Nome_Produto
FROM Produto
WHERE Produto.id_produto NOT IN (
SELECT Item_Nota_Fiscal.id_produto FROM Item_Nota_Fiscal);
```

	Nome_Produto
1	Antena



- **h. Listar o histórico de utilização de um determinado .**

```
--H - Listar o histórico de utilização de um determinado <veículo>.
DECLARE @id_veiculo INT = 1;
SELECT Locacao.data AS Data_Utilizacao, (SELECT Vendedor.nome FROM Vendedor WHERE Vendedor.id_vendedor = Locacao.id_vendedor) AS Nome_Vendedor
FROM Locacao WHERE Locacao.id_veiculo = @id_veiculo;
```

91 %

Resultados Mensagens

	Data_Utilizacao	Nome_Vendedor
1	2023-10-01	Mariana

- **i. A quantidade de itens de cada nota fiscal.**

```
--I - A quantidade de itens de cada nota fiscal
SELECT Nota_Fiscal.id_nota_fiscal AS Nota_Fiscal_ID, (SELECT SUM(Item_Nota_Fiscal.quant_vendida)
FROM Item_Nota_Fiscal WHERE Item_Nota_Fiscal.id_nota_fiscal = Nota_Fiscal.id_nota_fiscal) AS Total_Itens
FROM Nota_Fiscal;
```

91 %

Resultados Mensagens

	Nota_Fiscal_ID	Total_Itens
1	1	3
2	2	1
3	3	5
4	4	10
5	5	2

## f) Código

--Criando Banco de Dados

```
CREATE DATABASE BD_IndustriaBeleza;
```

```
USE BD_IndustriaBeleza;
```

--Criação das tabelas

-- Tabela Regiao

```
CREATE TABLE Regiao (
    codigo INT PRIMARY KEY,
    nome VARCHAR(100),
    descricao TEXT
);
```

-- Tabela Ponto\_Estrategico

```
CREATE TABLE Ponto_Estrategico (
    id INT PRIMARY KEY,
    nome VARCHAR(100),
    endereco TEXT,
    codigo_regiao INT,
    FOREIGN KEY (codigo_regiao) REFERENCES Regiao(codigo)
);
```

-- Tabela Cliente

```
CREATE TABLE Cliente (
    id_cliente INT PRIMARY KEY,
    nome VARCHAR(100),
    endereco TEXT,
    telefone VARCHAR(15)
);
```

```

-- Tabela Vendedor
CREATE TABLE Vendedor (
  id_vendedor INT PRIMARY KEY,
  nome VARCHAR(100),
  telefone VARCHAR(15),
  email VARCHAR(100),
  codigo_da_regiao INT,
  FOREIGN KEY (codigo_da_regiao) REFERENCES Regiao(codigo)
);

-- Tabela Veiculo
CREATE TABLE Veiculo (
  id_veiculo INT PRIMARY KEY,
  placa VARCHAR(10),
  modelo VARCHAR(50),
  ano INT,
  id_vendedor INT,
  FOREIGN KEY (id_vendedor) REFERENCES Vendedor(id_vendedor)
);

-- Tabela Locacao
CREATE TABLE Locacao (
  id_veiculo INT,
  id_vendedor INT,
  data DATE,
  PRIMARY KEY (id_veiculo, id_vendedor, data),
  FOREIGN KEY (id_veiculo) REFERENCES Veiculo(id_veiculo),
  FOREIGN KEY (id_vendedor) REFERENCES Vendedor(id_vendedor)
);

-- Tabela Produto
CREATE TABLE Produto (
  id_produto INT PRIMARY KEY,
  nome VARCHAR(100),
  descricao TEXT,
  preco DECIMAL(10, 2),
  quant_estoque INT
);

-- Tabela Nota_Fiscal
CREATE TABLE Nota_Fiscal (
  id_nota_fiscal INT PRIMARY KEY,
  id_cliente INT,
  data DATE,
  id_vendedor INT,
  FOREIGN KEY (id_cliente) REFERENCES Cliente(id_cliente),
  FOREIGN KEY (id_vendedor) REFERENCES Vendedor(id_vendedor)
);

```

```
-- Tabela Item_Nota_Fiscal
CREATE TABLE Item_Nota_Fiscal (
  id INT PRIMARY KEY,
  id_nota_fiscal INT,
  id_produto INT,
  quant_vendida INT,
  FOREIGN KEY (id_nota_fiscal) REFERENCES Nota_Fiscal(id_nota_fiscal),
  FOREIGN KEY (id_produto) REFERENCES Produto(id_produto)
);
```

```
-- Inserts para a tabela Regiao
INSERT INTO Regiao (codigo, nome, descricao) VALUES
(1, 'Região Norte', 'Estados do norte do Brasil'),
(2, 'Região Sul', 'Estados do sul do Brasil'),
(3, 'Região Sudeste', 'Estados do sudeste do Brasil'),
(4, 'Região Centro-Oeste', 'Estados do centro-oeste do Brasil'),
(5, 'Região Nordeste', 'Estados do nordeste do Brasil');
SELECT * FROM Regiao;
```

```
-- Inserts para a tabela Ponto_Estrategico
INSERT INTO Ponto_Estrategico (id, nome, endereco, codigo_regiao) VALUES
(1, 'Ponto de Vendas Amazonas', 'Av. Manaus, 200, Manaus', 1),
(2, 'Ponto Sul', 'Rua Paraná, 45, Curitiba', 2),
(3, 'Ponto de Acesso São Paulo', 'Rua Paulista, 100, São Paulo', 3),
(4, 'Centro Oeste Atacadista', 'Av. Goiás, 75, Goiânia', 4),
(5, 'Nordeste Center', 'Rua Bahia, 88, Salvador', 5);
SELECT * FROM Ponto_Estrategico;
```

```
-- Inserts para a tabela Cliente
INSERT INTO Cliente (id_cliente, nome, endereco, telefone) VALUES
(1, 'Odair', 'Rua das Flores, 123, Brasília', '(61) 91234-5678'),
(2, 'Juliana', 'Avenida Brasil, 456, Rio de Janeiro', '(21) 98765-4321'),
(3, 'Giullia', 'Rua dos Limoeiros, 789, Belo Horizonte', '(31) 99876-5432'),
(4, 'Larissa', 'Praça das Águas, 321, São Paulo', '(11) 93456-7890'),
(5, 'Alexandre', 'Alameda das Palmeiras, 654, Porto Alegre', '(51) 91234-0987');
SELECT * FROM Cliente;
```

```
-- Inserts para a tabela Vendedor
INSERT INTO Vendedor (id_vendedor, nome, telefone, email, codigo_da_regiao)
VALUES
(1, 'Mariana', '(61) 90000-0001', 'mariana@gmail.com', 1),
(2, 'Laura', '(21) 90000-0002', 'laura@gmail.com', 2),
(3, 'Luiza', '(31) 90000-0003', 'luiza@gmail.com', 3),
(4, 'Lucia', '(11) 90000-0004', 'lucia@gmail.com', 4),
(5, 'João', '(51) 90000-0005', 'joao@gmail.com', 5);
SELECT * FROM Vendedor;
```

-- Inserts para a tabela Veiculo

```
INSERT INTO Veiculo (id_veiculo, placa, modelo, ano, id_vendedor) VALUES  
(1, 'ABC-1A2', 'Fusca', 1974, 1),  
(2, 'XYZ-3B4', 'Civic', 2020, 2),  
(3, 'DEF-5C6', 'Corolla', 2021, 3),  
(4, 'GHI-7D8', 'Fiesta', 2022, 4),  
(5, 'JKL-9E0', 'Onix', 2023, 5);  
SELECT * FROM Veiculo;
```

-- Inserts para a tabela Locacao

```
INSERT INTO Locacao (id_veiculo, id_vendedor, data) VALUES  
(1, 1, '2023-10-01'),  
(2, 2, '2023-10-05'),  
(3, 3, '2023-10-10'),  
(4, 4, '2023-10-15'),  
(5, 5, '2023-10-20');  
SELECT * FROM Locacao;
```

-- Inserts para a tabela Produto

```
INSERT INTO Produto (id_produto, nome, descricao, preco, quant_estoque)  
VALUES  
(1, 'Rádio', 'Rádio para carro', 12.99, 150),  
(2, 'Capas para banco', 'Protetores para os bancos do carro', 5.50, 200),  
(3, 'Câmeras de ré', 'Câmeras que ajudam na visualização ao dar marcha à ré.',  
3.00, 300),  
(4, 'Suportes para celular', 'Permitem fixar o celular em um local acessível.', 1.50,  
1000),  
(5, 'Organizadores', 'Organizadores para carro', 4.75, 250),  
(6, 'Antena', 'Antena', 3.50, 100);  
SELECT * FROM Produto;
```

-- Inserts para a tabela Nota\_Fiscal

```
INSERT INTO Nota_Fiscal (id_nota_fiscal, id_cliente, data, id_vendedor) VALUES  
(1, 1, '2023-10-02', 1),  
(2, 2, '2023-10-06', 2),  
(3, 3, '2023-10-11', 3),  
(4, 4, '2023-10-16', 4),  
(5, 5, '2023-10-21', 5);  
SELECT * FROM Nota_Fiscal;
```

-- Inserts para a tabela Item\_Nota\_Fiscal

```
INSERT INTO Item_Nota_Fiscal (id, id_nota_fiscal, id_produto, quant_vendida)  
VALUES  
(1, 1, 1, 3),  
(2, 2, 2, 1),  
(3, 3, 3, 5),  
(4, 4, 4, 10),  
(5, 5, 5, 2);  
SELECT * FROM Item_Nota_Fiscal;
```

--A – Listar todos os pontos estratégicos de cada região.

```
SELECT nome AS Nome_Ponto_Estrategico, (SELECT nome FROM Regiao WHERE  
Regiao.codigo = Ponto_Estrategico.codigo_regiao) AS Nome_Regiao  
FROM Ponto_Estrategico;
```

--B – Listar os nomes das regiões cadastradas.

```
SELECT nome AS Nome_Regiao  
FROM Regiao;
```

--C – Listar todos os vendedores e quais veículos que eles utilizaram no último mês.

```
SELECT V.nome AS Nome_Vendedor, VE.modelo AS Modelo_Veiculo, L.data AS  
Data_Utilizacao  
FROM Vendedor V  
JOIN Veiculo VE ON V.id_vendedor = VE.id_vendedor  
JOIN Locacao L ON VE.id_veiculo = L.id_veiculo;
```

--D – Listar todos os vendedores responsáveis por cada região.

```
SELECT (SELECT Regiao.nome FROM Regiao WHERE Regiao.codigo =  
Vendedor.codigo_da_regiao) AS Nome_Regiao, Vendedor.nome AS  
Nome_Vendedor  
FROM Vendedor;
```

--E – Todos os produtos vendidos por um determinado <vendedor>.

```
DECLARE @id_vendedor INT = 2; -- ID de Laura  
SELECT V.nome AS Nome_Vendedor, P.nome AS Nome_Produto  
FROM Produto P  
JOIN Item_Nota_Fiscal INF ON P.id_produto = INF.id_produto  
JOIN Nota_Fiscal NF ON INF.id_nota_fiscal = NF.id_nota_fiscal  
JOIN Vendedor V ON NF.id_vendedor = V.id_vendedor  
WHERE NF.id_vendedor = @id_vendedor;
```

--F – Todos os vendedores que venderam um determinado <produto>.

```
DECLARE @id_produto INT = 1;  
SELECT P.nome AS Nome_Produto, V.nome AS Nome_Vendedor  
FROM Vendedor V  
JOIN Nota_Fiscal NF ON V.id_vendedor = NF.id_vendedor  
JOIN Item_Nota_Fiscal INF ON NF.id_nota_fiscal = INF.id_nota_fiscal  
JOIN Produto P ON INF.id_produto = P.id_produto  
WHERE INF.id_produto = @id_produto;
```

--G – Todos os produtos que ainda não foram vendidos.

```
SELECT Produto.nome AS Nome_Produto  
FROM Produto  
WHERE Produto.id_produto NOT IN (  
SELECT Item_Nota_Fiscal.id_produto FROM Item_Nota_Fiscal);
```

--H – Listar o histórico de utilização de um determinado <veículo>.

DECLARE @id\_veiculo INT = 1;

SELECT Locacao.data AS Data\_Utilizacao,(SELECT Vendedor.nome FROM Vendedor WHERE Vendedor.id\_vendedor = Locacao.id\_vendedor) AS Nome\_Vendedor

FROM Locacao WHERE Locacao.id\_veiculo = @id\_veiculo;

--I - A quantidade de itens de cada nota fiscal

SELECT Nota\_Fiscal.id\_nota\_fiscal AS Nota\_Fiscal\_ID, (SELECT SUM(Item\_Nota\_Fiscal.quant\_vendida)

FROM Item\_Nota\_Fiscal WHERE Item\_Nota\_Fiscal.id\_nota\_fiscal = Nota\_Fiscal.id\_nota\_fiscal) AS Total\_Itens

FROM Nota\_Fiscal;



**Professora, muito obrigada pelos ensinamentos! Aprendi demais com você e foi ótimo ter você como professora nesses últimos 3 anos!**

Giovana Marsigli Rodrigues – 3DS AMS

