**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA ENSINO TÉCNICO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**AMS**

DANILO SANTOS SOARES

**DESENVOLVIMENTO DA ATIVIDADE** “Utilização de Inner-join Banco CD, Departamento e Garagem”

**São Paulo**

**2024**

**Exercícios de SQL**

1. **-** Crie o banco de dados com as respectivas tabelas:

|  |  |
| --- | --- |
| **Cds** | **Musicas** |
| **Codigo** **(PK)**  Nome\_cd  DataCompra  ValorPago  LocalCompra  Album (Sim ou Não) | **CodigoCD (FK)**  Numero\_ musica  Nome\_musica  Artista  Tempo |

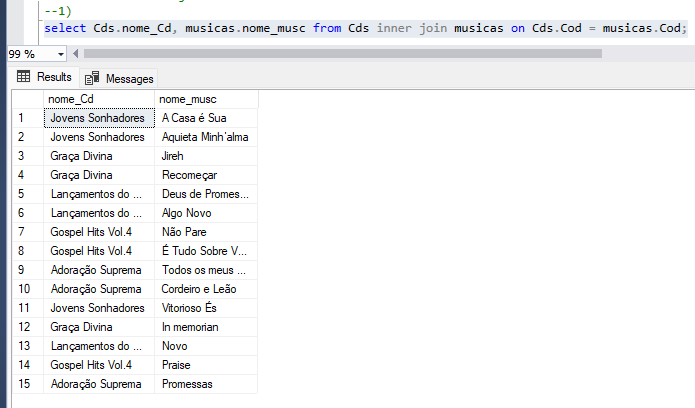
1. – Utilizar os comandos Selects respectivos para efetuar as consultas abaixo:

1. **(Inner Join)** Mostrar o nome do Cd e o nome das músicas de todos Cds

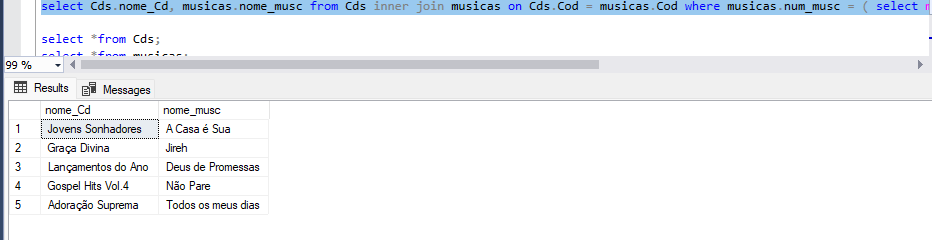
1. **(Inner Join)**Mostre o nome do cd e o nome da primeira música de cada cd

**Prints Cds inner-join:**

1. **(Inner Join)** Mostrar o nome do Cd e o nome das músicas de todos Cds



1. **(Inner Join)**Mostre o nome do cd e o nome da primeira música de cada cd



**Código Cds:**

create database lojaCD;

use lojaCD;

create table Cds(

Cod int unique,

nome\_Cd varchar(50),

dataCompra date,

ValorPago float,

LocalCompra varchar(40),

Album varchar(30),

primary key(Cod)

);

create table musicas(

Cod int,

num\_musc int,

nome\_musc varchar (50),

Artista varchar (50),

Tempo time,

foreign key (Cod) references Cds(cod),

);

INSERT INTO Cds (Cod, nome\_Cd, dataCompra, ValorPago, LocalCompra, Album) VALUES

(1, 'Jovens Sonhadores', '2024-04-01', 15.99, 'Loja de Música', 'Sim'),

(2, 'Graça Divina', '2024-03-15', 12.50, 'Loja Online Gospel', 'Sim'),

(3, 'Lançamentos do Ano', '2024-04-03', 20.00, 'Super music', 'Sim'),

(4, 'Gospel Hits Vol.4', '2024-01-25', 18.75, 'Lojamusic', 'Não'),

(5, 'Adoração Suprema', '2024-02-10', 22.00, 'OnlineMusic', 'Não');

INSERT INTO musicas (Cod, num\_musc, nome\_musc, Artista, Tempo) VALUES

(1, 1, 'A Casa é Sua', 'Casa Worship', '00:05:30'),

(1, 2, 'Aquieta Minh’alma', 'Ministério Zoe', '00:06:40'),

(2, 3, 'Jireh', 'Elevation Worship & Maverick City', '00:09:59'),

(2, 4, 'Recomeçar', 'Juliano Son', '00:04:22'),

(3, 5, 'Deus de Promessas', 'Davi Sacer', '00:05:15'),

(3, 6, 'Algo Novo', 'Kemuel, Lukas Agustinho', '00:04:55'),

(4, 7, 'Não Pare', 'Midian Lima', '00:05:10'),

(4, 8, 'É Tudo Sobre Você', 'Ministério Morada', '00:06:05'),

(5, 9, 'Todos os meus dias', 'Gabriel Guedes', '00:03:23'),

(5, 10, 'Cordeiro e Leão', 'Isaque Valadão', '00:05:34'),

(1, 11, 'Vitorioso És', 'Gabriel Guedes', '00:06:00'),

(2, 12, 'In memorian', 'Gabriel Guedes', '00:05:07'),

(3, 13, 'Novo', 'Lagoinha Orlando', '00:04:14'),

(4, 14, 'Praise', 'Elevation Worship', '00:05:04'),

(5, 15, 'Promessas', 'Sarah Beatriz', '00:06:25');

--1)

select\*from Cds;

--2)

select nome\_Cd, DataCompra from Cds Order by nome\_Cd;

--3)

select nome\_Cd, DataCompra from Cds order by dataCompra desc;

--4)

select nome\_Cd, DataCompra, ValorPago from Cds order by ValorPago desc;

--5)

select Sum(ValorPago) as ValorTotalGasto from Cds;

--6)

select \*from musicas where Cod = 1

--7)

select nome\_musc, Artista from musicas;

--8)

select num\_musc, nome\_musc, Tempo from musicas where Cod = 5 order by num\_musc;

--9)

select count(num\_musc) from musicas;

--10)

select nome\_musc from musicas where Artista='José Pedro';

--ou

select nome\_musc from musicas where Artista='Gabriel Guedes';

--11)

select nome\_Cd from Cds where LocalCompra='submárino';

--ou

select nome\_Cd from Cds where LocalCompra='Super music';

--12)

select nome\_musc from musicas order by nome\_musc asc;

--13)

select\* from Cds where Album = 'Sim';

--14)

select avg(ValorPago)from Cds;

--Comando de inner join

--1)

select Cds.nome\_Cd, musicas.nome\_musc from Cds inner join musicas on Cds.Cod = musicas.Cod;

--2)

select Cds.nome\_Cd, musicas.nome\_musc from Cds inner join musicas on Cds.Cod = musicas.Cod where musicas.num\_musc=1 ;

select Cds.nome\_Cd, musicas.nome\_musc from Cds inner join musicas on Cds.Cod = musicas.Cod where musicas.num\_musc = ( select min(num\_musc)from musicas where Cod = Cds.Cod);

select \*from Cds;

select \*from musicas;

drop table Cds;

drop table musicas;

**Exercícios de SQL**

**A-** Crie o banco de dados com as respectivas tabelas:

|  |  |
| --- | --- |
| **Funcionarios** | **Departamentos** |
| Codigo\_func (PK)  NomeFunc  SobreNome  -- DataNasci  CPF  RG  Endereco  CEP  Cidade  Fone  CodigoDepartamento,(FK)  Funcao  Salario | Codigo\_depto (PK)  NomeDepto |

– Utilizar os comandos Selects respectivos para efetuar as consultas abaixo:

- **Select com Inner Join**

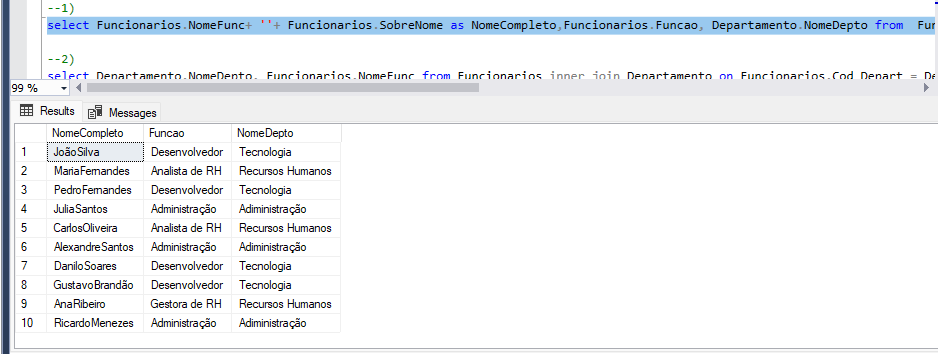
1. - Liste o nome completo do funcionário, o nome do departamento e a função de todos os funcionários

1. - Liste os nomes dos departamentos e dos funcionários que têm a função de supervisor

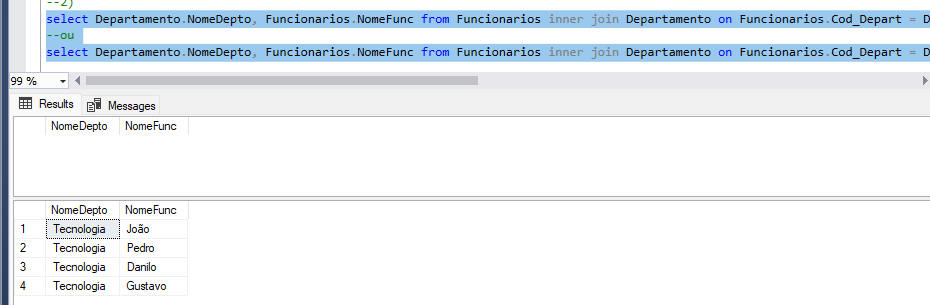
# **3** - Liste o nome do departamento e nome completo do funcionário ordenados por departamento e Sobrenome do funcionário

**Prints Departamentos inner-join**

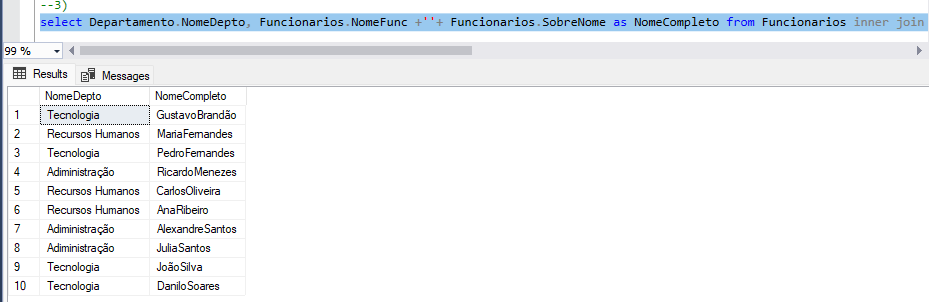
**1- Liste o nome completo do funcionário, o nome do departamento e a função de todos os funcionários**



**2- Liste os nomes dos departamentos e dos funcionários que têm a função de supervisor**



# **3- Liste o nome do departamento e nome completo do funcionário ordenados por departamento e Sobrenome do funcionário**



**Código Departamento:**

create database Departamentos;

use Departamentos;

create table Departamento(

Cod\_Depart int identity,

NomeDepto varchar(30),

primary key(Cod\_Depart),

);

create table Funcionarios(

Cod\_Func int identity,

NomeFunc Varchar(14),

SobreNome varchar(30),

DataNasc date,

CPF varchar(14),

RG varchar(17),

Endereco varchar(30),

CEP varchar(21),

Cidade varchar(20),

Fone varchar(15),

Cod\_Depart int,

Funcao varchar(50),

Salario float,

primary key (Cod\_Func),

foreign key(Cod\_Depart) references Departamento(Cod\_Depart)

);

INSERT INTO Departamento(NomeDepto) VALUES

('Tecnologia'),

('Recursos Humanos'),

('Adiministração');

insert into Funcionarios (NomeFunc, SobreNome, DataNasc, CPF, RG, Endereco, CEP, Cidade, Fone, Cod\_Depart, Funcao, Salario) values

('João', 'Silva', '1985-06-15', '123.456.789-09', 'MG-10.987.654', 'Rua das Flores, 123', '31200-000', 'Belo Horizonte', '(11) 94524-8468', 1, 'Desenvolvedor', 4500.00),

('Maria', 'Fernandes', '1990-08-25', '987.654.321-09', 'SP-20.876.543', 'Av. Paulista, 1800', '04567-000', 'São Paulo', '(11) 95612-4582', 2, 'Analista de RH', 3500.00),

('Pedro', 'Fernandes', '2000-05-18', '937.674.211-10', 'SP-15.926.133', 'Rua Augusta, 245', '17229-000', 'São Paulo', '(11) 96482-8464', 1, 'Desenvolvedor', 4500.00),

('Julia', 'Santos', '1999-07-13', '953.874.231-04', 'SP-21.347.286', 'Av. Ibirapuera, 405', '34575-480', 'São Paulo', '(11) 99587-4582', 3, 'Administração', 4500.00),

('Carlos', 'Oliveira', '1980-03-20', '948.495.864-08', 'SP-12.954.846', 'Av. Brigadeiro, 1572', '93485-846', 'São Paulo', '(11) 97561-6482', 2, 'Analista de RH', 5000.00),

('Alexandre', 'Santos', '2002-02-14', '485.845.548-23', 'SP-85.478.785', 'Rua Liberdade, 190', '45513-845', 'São Paulo', '(11) 99845-7456', 3, 'Administração', 2300.00),

('Danilo', 'Soares', '2005-06-01', '953.874.231-04', 'SP-21.347.286', 'Rua dos Pinheiros, 506', '34575-480', 'São Paulo', '(11) 98452-8464', 1, 'Desenvolvedor', 2500.00),

('Gustavo', 'Brandão', '2006-04-13', '253.068.256-09', 'SP-41.312.186', 'Av. Rebouças, 3000', '48135-754', 'São Paulo', '(11) 98442-8945', 1, 'Desenvolvedor', 2500.00),

('Ana', 'Ribeiro', '1992-10-07', '222.333.444-55', 'SP-15.666.777', 'Av. Ana Costa, 475', '11000-000', 'Santos', '(13) 98234-5678', 2, 'Gestora de RH', 4800.00),

('Ricardo', 'Menezes', '1988-04-25', '555.666.777-88', 'RJ-22.333.444', 'Rua Visconde de Pirajá, 414', '22090-000', 'Rio de Janeiro', '(21) 97531-8899', 3, 'Administração', 7000.00);

--1)

select NomeFunc, SobreNome from Funcionarios order by SobreNome;

--2)

select \* from Funcionarios order by Cidade;

--3)

select \* from Funcionarios where Salario > 1000 order by NomeFunc, SobreNome;

--4)

select DataNasc, NomeFunc from Funcionarios order by DataNasc desc;

--5)

select SobreNome, NomeFunc, Cidade,Endereco ,Fone from Funcionarios;

--6)

select sum(Salario) as TotalPagamento from Funcionarios;

--7)

select NomeFunc + ' ' + SobreNome as NomeCompleto, Salario, Funcao from Funcionarios order by Funcao;

--8)

select \* from Funcionarios where Funcao = 'Supervisor';

--ou

select \* from Funcionarios where Funcao = 'Desenvolvedor';

--9)

select Count(\*) as TotalFuncionarios from Funcionarios;

--10)

select avg(Salario) as MédiaPago from Funcionarios;

--11)

select \* from Funcionarios where Funcao='Telefonista' and Cidade=' Recife';

--ou

select \*from Funcionarios where Funcao='Administração' and Cidade='São paulo';

--Selects com inner join

--1)

select Funcionarios.NomeFunc+ ''+ Funcionarios.SobreNome as NomeCompleto,Funcionarios.Funcao, Departamento.NomeDepto from Funcionarios inner join Departamento on Funcionarios.Cod\_Depart = Departamento.Cod\_Depart;

--2)

select Departamento.NomeDepto, Funcionarios.NomeFunc from Funcionarios inner join Departamento on Funcionarios.Cod\_Depart = Departamento.Cod\_Depart where funcao ='supervisor';

--ou

select Departamento.NomeDepto, Funcionarios.NomeFunc from Funcionarios inner join Departamento on Funcionarios.Cod\_Depart = Departamento.Cod\_Depart where funcao ='Desenvolvedor';

--3)

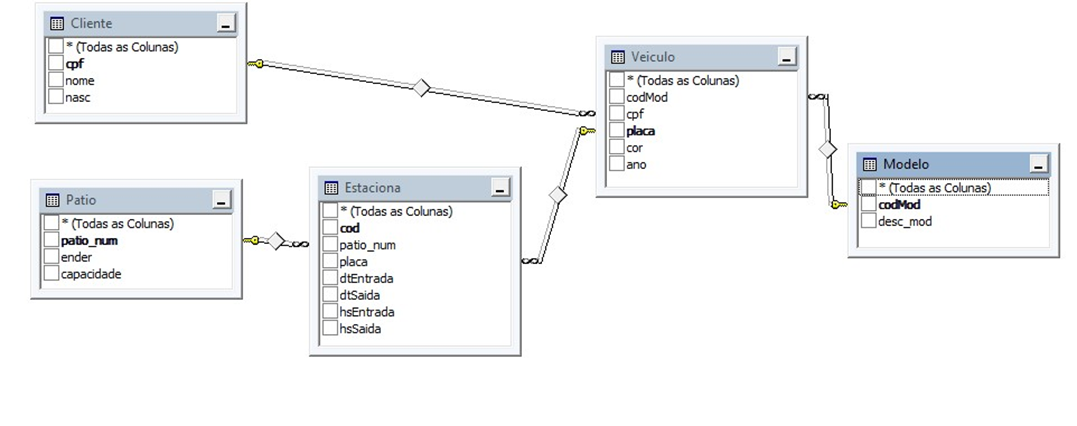
select Departamento.NomeDepto, Funcionarios.NomeFunc +''+ Funcionarios.SobreNome as NomeCompleto from Funcionarios inner join Departamento on Funcionarios.Cod\_Depart = Departamento.Cod\_Depart order by SobreNome ,Departamento.NomeDepto ;

select \*from Departamento;

select \* from Funcionarios

drop table Funcionarios;

drop table Departamento;

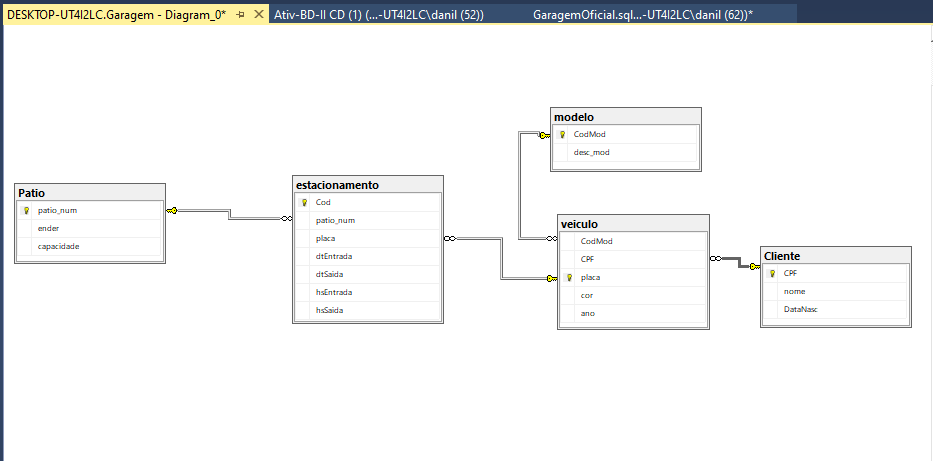


1. Criar esta base de dados com o nome de “**Garagem**”.
2. Inclua alguns registros coerentemente com os enunciados.
3. Desenvolva os “Inner Join´s” a seguir:

* 1. Exiba a placa e o nome dos donos de todos os veículos
  2. Exiba o CPF e o nome do cliente que possui o veículo de placa “JJJ-2020”
  3. Exiba a placa e a cor do veículo que possui o código de estacionamento 1
  4. Exiba o ano e a placa do veículo que possui o código de estacionamento 1
  5. Exiba a placa, o ano do veículo e a descrição de seu modelo, se ele possuir ano a
     1. partir de 2000
  6. Exiba o endereço, a data de entrada e de saída dos estacionamentos do veículo de placa “JEG-1010”
  7. Exiba a quantidade de vezes os veículos de cor verde estacionaram.
  8. Liste todos os clientes que possuem carro de modelo 1
  9. Liste as placas, os horários de entrada e saída dos veículos de cor verde
  10. Liste todos os estacionamentos do veículo de placa “JJJ-2020”
  11. Exiba o nome do cliente que possui o veículo cujo código do estacionamento é 2
  12. Exiba o CPF do cliente que possui o veículo cujo código do estacionamento é 3
  13. Exiba a descrição do modelo do veículo cujo código do estacionamento é 2
  14. Exiba a placa, o nome dos donos e a descrição dos modelos de todos os veículos.

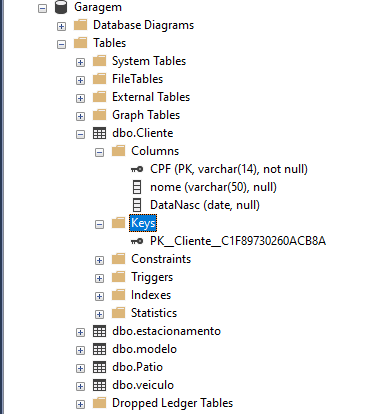
**Prints:**

**MER:**

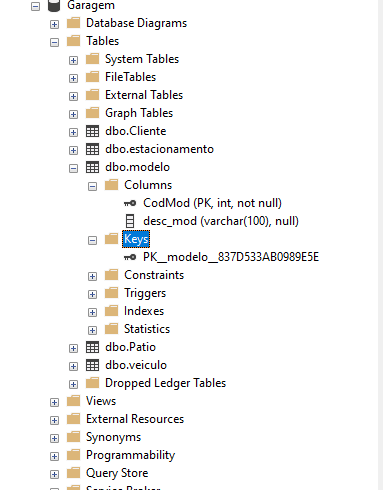


**Estrutura:**

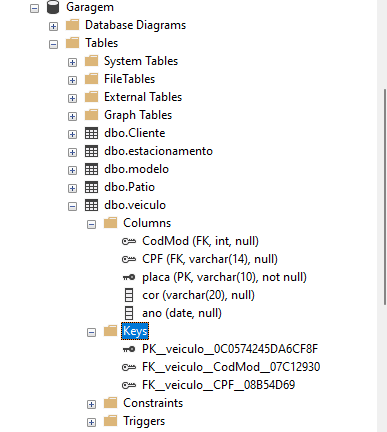
**Cliente:**



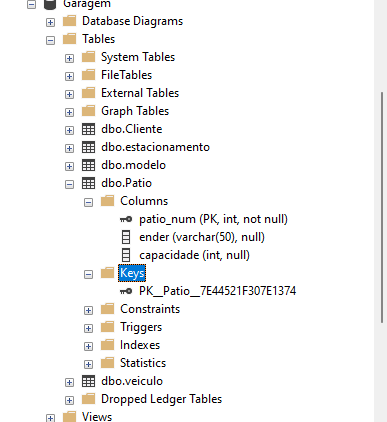
**Modelo:**



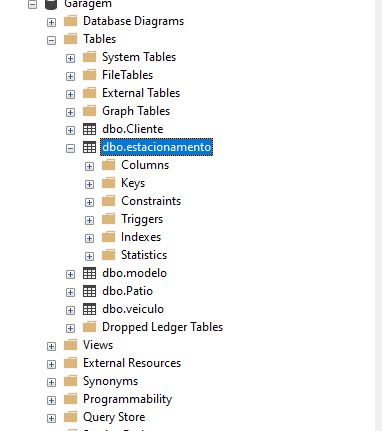
**Veículo:**



**Patio:**

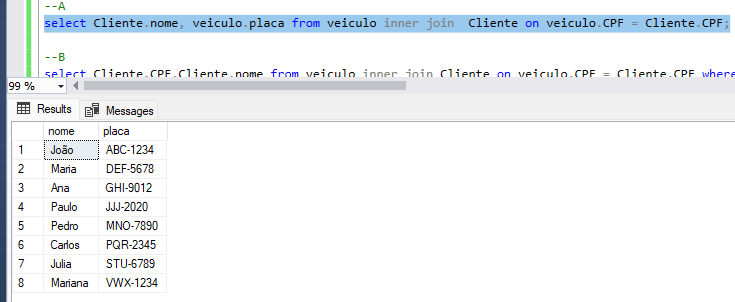


**Estacionamento:**

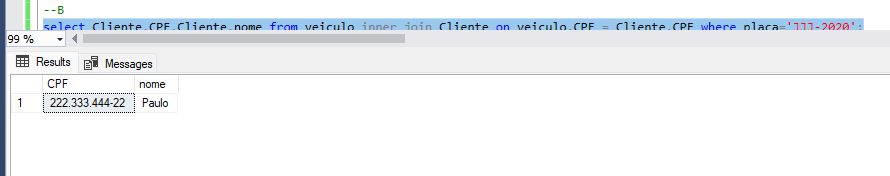


**Questões:**

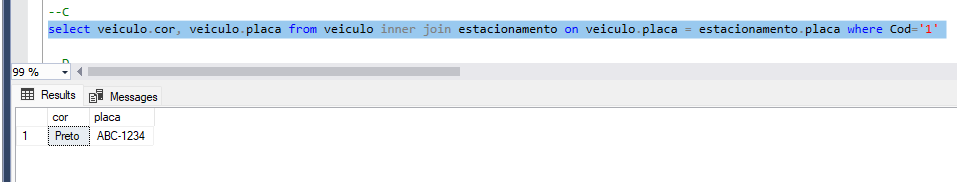
* 1. **Exiba a placa e o nome dos donos de todos os veículos**



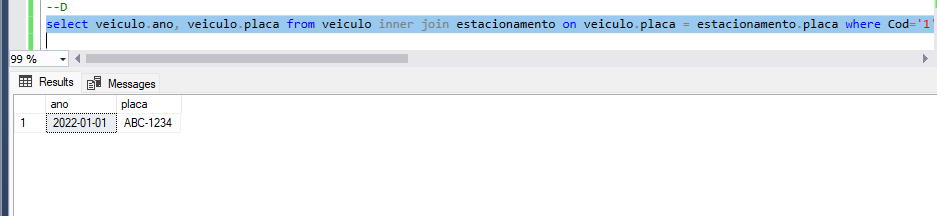
* 1. **Exiba o CPF e o nome do cliente que possui o veículo de placa “JJJ-2020”**



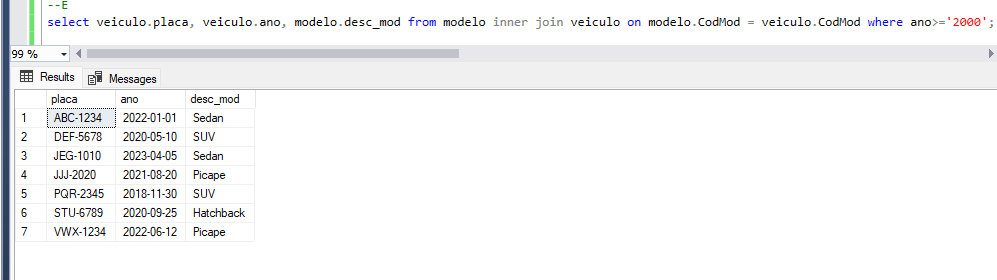
* 1. **Exiba a placa e a cor do veículo que possui o código de estacionamento 1**



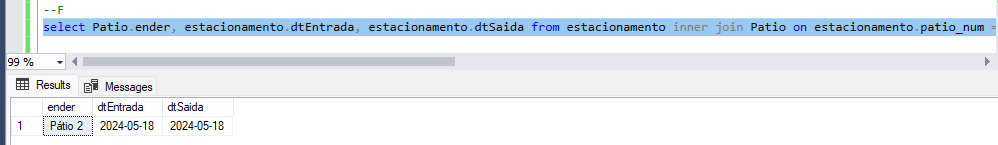
* 1. **Exiba o ano e a placa do veículo que possui o código de estacionamento 1**



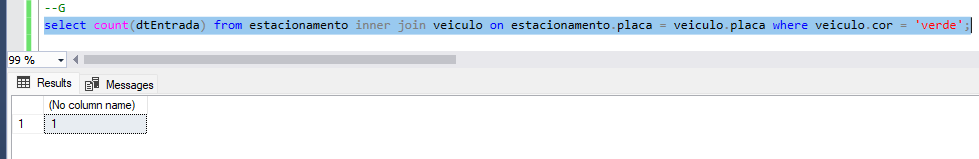
* 1. **Exiba a placa, o ano do veículo e a descrição de seu modelo, se ele possuir ano a partir de 2000**



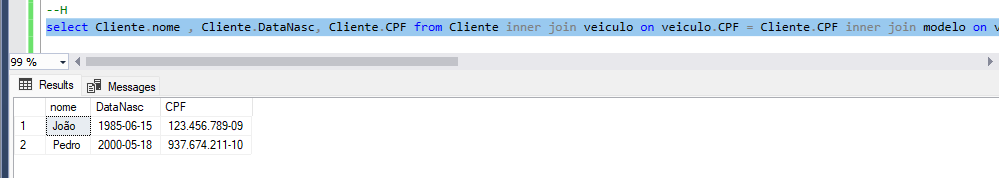
* 1. **Exiba o endereço, a data de entrada e de saída dos estacionamentos do veículo de placa “JEG-1010”**



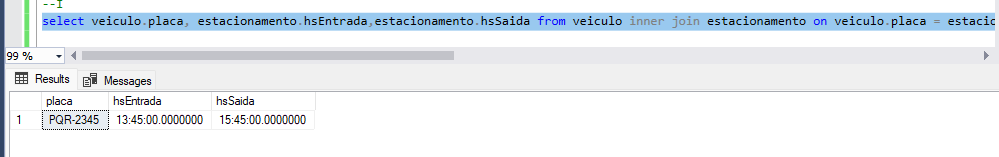
* 1. **Exiba a quantidade de vezes os veículos de cor verde estacionaram.**



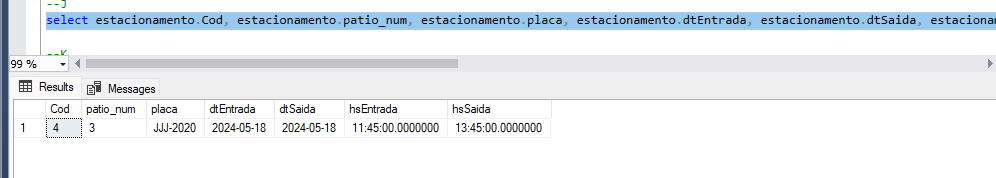
* 1. **Liste todos os clientes que possuem carro de modelo 1**



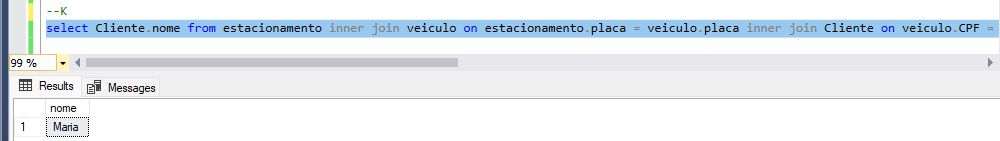
* 1. **Liste as placas, os horários de entrada e saída dos veículos de cor verde**



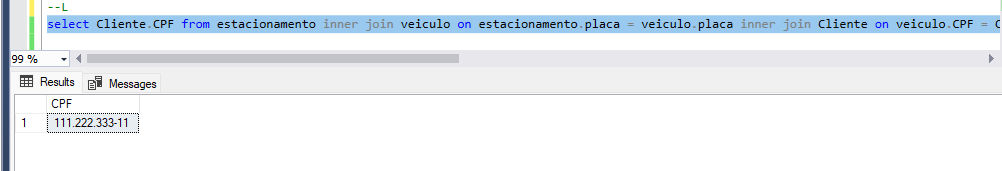
* 1. **Liste todos os estacionamentos do veículo de placa “JJJ-2020”**



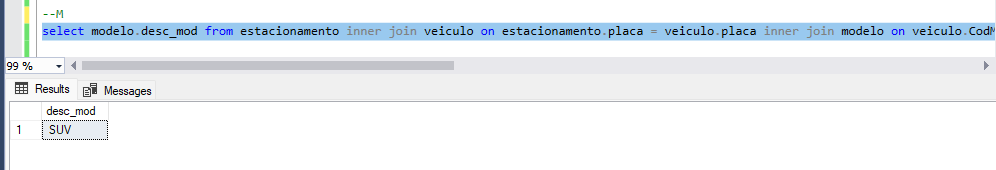
* 1. **Exiba o nome do cliente que possui o veículo cujo código do estacionamento é 2**



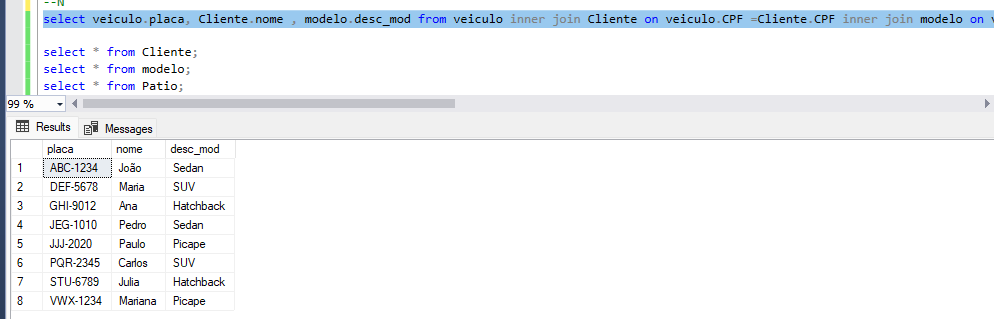
* 1. **Exiba o CPF do cliente que possui o veículo cujo código do estacionamento é 3**



* 1. **Exiba a descrição do modelo do veículo cujo código do estacionamento é 2**



* 1. **Exiba a placa, o nome dos donos e a descrição dos modelos de todos os veículos.**



**Código:**

CREATE DATABASE Garagem;

USE Garagem;

create table Cliente (

CPF varchar(14) ,

nome varchar(50),

DataNasc date,

Primary key (CPF)

);

create table modelo (

CodMod int,

desc\_mod varchar(100),

Primary key (CodMod)

);

create table Patio (

patio\_num int,

ender varchar(50),

capacidade int,

Primary key (patio\_num)

);

create table veiculo (

CodMod int ,

CPF varchar(14),

placa varchar(10) ,

cor varchar(20),

ano date,

Primary key (placa),

foreign key (CodMod) references modelo(CodMod),

foreign key (CPF) references Cliente(CPF)

);

create table estacionamento (

Cod int,

patio\_num int ,

placa varchar(10) ,

dtEntrada date,

dtSaida date,

hsEntrada time(7),

hsSaida time(7),

primary key (Cod),

foreign key (patio\_num) references Patio(patio\_num),

foreign key (placa) references veiculo(placa)

);

insert into Cliente (CPF, nome, DataNasc) values

('111.222.333-11', 'Ana', '1995-11-30'),

('123.456.789-09', 'João', '1985-06-15'),

('222.333.444-22', 'Paulo', '1988-09-22'),

('333.444.555-33', 'Mariana', '1975-04-10'),

('937.674.211-10', 'Pedro', '2000-05-18'),

('948.495.864-08', 'Carlos', '1980-03-20'),

('953.874.231-04', 'Julia', '1999-07-13'),

('987.654.321-09', 'Maria', '1990-08-25');

insert into modelo (CodMod, desc\_mod) values

(1, 'Sedan'),

(2, 'SUV'),

(3, 'Hatchback'),

(4, 'Picape'),

(5, 'Crossover'),

(6, 'Minivan');

insert into Patio (patio\_num, ender, capacidade)values

(1, 'Pátio 1', 50),

(2, 'Pátio 2', 70),

(3, 'Pátio 3', 60),

(4, 'Pátio 4', 80),

(5, 'Pátio 5', 90);

INSERT INTO veiculo(CodMod, CPF, placa, cor, ano) VALUES

(1, '123.456.789-09', 'ABC-1234', 'Preto', '2022-01-01'),

(2, '987.654.321-09', 'DEF-5678', 'Branco', '2020-05-10'),

(3, '111.222.333-11', 'GHI-9012', 'Azul', '1998-12-15'),

(4, '222.333.444-22', 'JJJ-2020', 'Vermelho', '2021-08-20'),

(1, '937.674.211-10', 'JEG-1010', 'Prata', '2023-04-05'),

(2, '948.495.864-08', 'PQR-2345', 'Verde', '2018-11-30'),

(3, '953.874.231-04', 'STU-6789', 'Cinza', '2020-09-25'),

(4, '333.444.555-33', 'VWX-1234', 'Amarelo', '2022-06-12');

INSERT INTO estacionamento (cod, patio\_num, placa, dtEntrada, dtSaida, hsEntrada, hsSaida) VALUES

(1,1, 'ABC-1234', '2024-05-18', '2024-05-18', '09:30:00', '11:30:00'),

(2,1, 'DEF-5678', '2024-05-18', '2024-05-18', '10:15:00', '12:15:00'),

(3,2, 'GHI-9012', '2024-05-18', '2024-05-18', '11:00:00', '13:00:00'),

(4,3, 'JJJ-2020', '2024-05-18', '2024-05-18', '11:45:00', '13:45:00'),

(5,2, 'JEG-1010', '2024-05-18', '2024-05-18', '13:00:00', '15:00:00'),

(6,4, 'PQR-2345', '2024-05-18', '2024-05-18', '13:45:00', '15:45:00'),

(7,5, 'STU-6789', '2024-05-18', '2024-05-18', '14:30:00', '16:30:00'),

(8,5, 'VWX-1234', '2024-05-18', '2024-05-18', '15:15:00', '17:15:00');

--A

select Cliente.nome, veiculo.placa from veiculo inner join Cliente on veiculo.CPF = Cliente.CPF;

--B

select Cliente.CPF,Cliente.nome from veiculo inner join Cliente on veiculo.CPF = Cliente.CPF where placa='JJJ-2020';

--C

select veiculo.cor, veiculo.placa from veiculo inner join estacionamento on veiculo.placa = estacionamento.placa where Cod='1'

--D

select veiculo.ano, veiculo.placa from veiculo inner join estacionamento on veiculo.placa = estacionamento.placa where Cod='1'

--E

select veiculo.placa, veiculo.ano, modelo.desc\_mod from modelo inner join veiculo on modelo.CodMod = veiculo.CodMod where ano>='2000';

--F

select Patio.ender, estacionamento.dtEntrada, estacionamento.dtSaida from estacionamento inner join Patio on estacionamento.patio\_num = Patio.patio\_num where estacionamento.placa ='JEG-1010';

--G

select count(dtEntrada) from estacionamento inner join veiculo on estacionamento.placa = veiculo.placa where veiculo.cor = 'verde';

--H

select Cliente.nome , Cliente.DataNasc, Cliente.CPF from Cliente inner join veiculo on veiculo.CPF = Cliente.CPF inner join modelo on veiculo.CodMod = modelo.CodMod where modelo.CodMod = '1'

--I

select veiculo.placa, estacionamento.hsEntrada,estacionamento.hsSaida from veiculo inner join estacionamento on veiculo.placa = estacionamento.placa where cor='verde';

--J

select estacionamento.Cod, estacionamento.patio\_num, estacionamento.placa, estacionamento.dtEntrada, estacionamento.dtSaida, estacionamento.hsEntrada, estacionamento.hsSaida from estacionamento inner join veiculo veiculo on estacionamento.placa = veiculo.placa where veiculo.placa = 'JJJ-2020';

--K

select Cliente.nome from estacionamento inner join veiculo on estacionamento.placa = veiculo.placa inner join Cliente on veiculo.CPF = Cliente.CPF where estacionamento.Cod = 2;

--L

select Cliente.CPF from estacionamento inner join veiculo on estacionamento.placa = veiculo.placa inner join Cliente on veiculo.CPF = Cliente.CPF where estacionamento.Cod = 3;

--M

select modelo.desc\_mod from estacionamento inner join veiculo on estacionamento.placa = veiculo.placa inner join modelo on veiculo.CodMod = modelo.CodMod where estacionamento.Cod = 2;

--N

select veiculo.placa, Cliente.nome , modelo.desc\_mod from veiculo inner join Cliente on veiculo.CPF =Cliente.CPF inner join modelo on veiculo.CodMod= modelo.CodMod;

select \* from Cliente;

select \* from modelo;

select \* from Patio;

select \* from veiculo;

select \* from estacionamento;

drop table estacionamento;

drop table veiculo;

drop table Patio;

drop table modelo;

drop table Cliente;

DROP DATABASE Garagem;