



Dênis de Souza Cordeiro - 202110235
Gabriel Fernando Zanda Gonçalves - 202110234
Ronald de Souza Galdino - 202110679
Wesley Henrique Santos Abreu - 202010834
Iago Carvalho Souto - 2021102030159

Trabalho Prático – Etapa 3 – Implementação em SQL

GCC214 - Introdução a Sistemas de Banco de Dados

Prof. Denilson Alves Pereira

Descrição do projeto:

(**Observação:** Os atributos estão em **negrito** apenas para facilitar a leitura)

Levantamento de dados para o sistema de controle de veículos em exposição para venda em uma rede de concessionárias chamada “EasyCars”. Para ter um controle completo da qualidade e procedência dos veículos sendo oferecidos aos clientes, essa rede faz questão de registrar o **CNPJ, Endereço Completo, Telefone e Número Atual de Veículos** das unidades concessionárias que recebem os veículos a serem expostos e vendidos exclusivamente nelas.

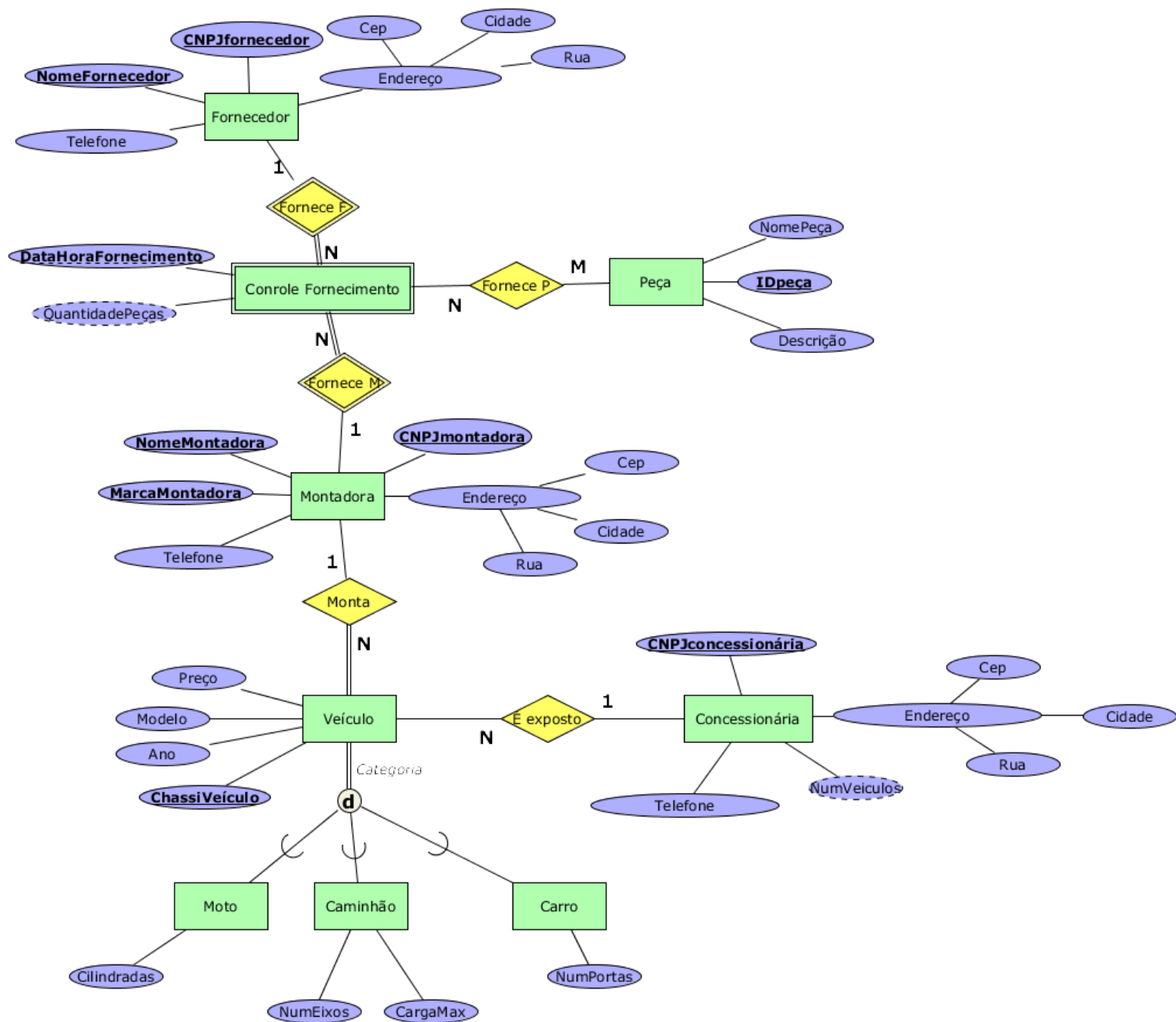
Para todos os veículos expostos para venda (têm-se apenas veículos das categorias: Carro, Moto, e Caminhão) é registrado o **Preço, Modelo, Ano de Fabricação e Número do Chassi**, e além de tais dados, se armazena também exclusivamente as **Cilindradas** para as motos, o **Número de Eixos e Carga Máxima** para os caminhões e o **Número de Portas** para os carros.

Os diversos veículos que são expostos nas concessionárias são montados por completo em montadoras (as montadoras fazem apenas veículos inteiros de uma marca específica), das quais se armazena o **Nome, CNPJ, Marca, Telefone e Endereço Completo**. Uma montadora pode receber diversas peças de fornecedores.

Como dito anteriormente, essa rede de concessionárias se preocupa muito com a qualidade e procedência de seus veículos, para um maior controle e organização, toda operação de fornecimento de peças que é realizada para as montadoras é cadastrada em um Controle de Fornecimento, que registra a **data hora** de um fornecimento proveniente de um fornecedor e a **Quantidade Total de Peças** envolvidas nele (mas um mesmo fornecedor pode fazer vários fornecimentos diferentes, para montadoras diferentes e com peças diferentes, contanto que em datas horas diferentes).

Dos fornecedores registra-se o **Nome, Telefone, CNPJ e Endereço Completo**. E das peças se armazena o **Nome, Identificador e Descrição**.

Diagrama ER:



Dicionário de Dados:

Tipo Entidade:	Veículo		
Descrição:	Conjunto de veículos que são criados		
Atributos			
Nome	Descrição	Domínio	Permite Nulo ? (S/N)
Preço	Valor do veículo	Real (6,2) positivo	N
Modelo	Nome do modelo do veículo	Texto (15)	N
Ano	Ano em que o veículo foi fabricado	Inteiro (4) Positivo	N
ChassiVeículo	Chassi de identificação do veículo	Texto (17)	N

Tipo Entidade:	Montadora		
Descrição:	Montadoras de veículos (recebem peças e constroem os veículos para enviar às concessionárias).		
Atributos			
Nome	Descrição	Domínio	Permite Nulo ? (S/N)
NomeMontadora	Nome da montadora	Texto (30)	N
MarcaMontadora	Marca da montadora	Texto (15)	N
Telefone	Telefone da montadora	Texto (10) - ((XX) XXXX-XXXX)	N
CNPJmontadora	Cadastro Nacional da Pessoa Jurídica	Texto (14) - (XX.XXX.XXX/XXXX-XX)	N
CEP	Código de endereçamento postal	Texto (8) - (XXXXX-XXX)	N
Cidade	Nome da cidade	Texto (30)	N

Tipo Entidade Fraca:	Controle fornecimento		
Descrição:	Controla o registro de fornecimento de peças para uma montadora por um fornecedor. Agindo como "histórico" de fornecimento.		
Atributos			
Nome	Descrição	Domínio	Permite Nulo ? (S/N)
DataHoraFornecimento	Data e Hora do fornecimento realizado de peças	Texto (14) - AAAA-MM-DD hh-mm-ss	N
QuantidadePeças	Quantidade total de peças fornecida nesse fornecimento	Inteiro (6) positivo	N

Tipo Entidade:	Peça		
Descrição:	Peça fabricada pelo fornecedor e que é enviada para a montadora.		
Atributos			
Nome	Descrição	Domínio	Permite Nulo ? (S/N)
NomePeça	Nome da peça	Texto (30)	N
IdPeça	Número identificador da peça	Inteiro (3) positivo	N
Descrição	Descrição sobre as características da peça	Texto (100)	N

Tipo Entidade:	Fornecedor		
Descrição:	Fornecedor que concede as peças para as montadoras.		
Atributos			
Nome	Descrição	Domínio	Permite Nulo ? (S/N)
NomeFornecedor	Nome do fornecedor	Texto (30)	N
Telefone	Telefone do fornecedor	Texto (10) - ((XX) XXXX-XXXX)	N
CNPJfornecedor	Cadastro Nacional da Pessoa Jurídica	Texto (14) - (XX.XXX.XXX/XXXX-XX)	N
CEP	Código de endereçamento postal	Texto (8) - (XXXXX-XXX)	N
Cidade	Cidade do fornecedor	Texto (30)	N

Tipo Entidade:	Concessionária		
Descrição:	Unidades de Concesssionária de veículos (onde os veículos ficam expostos para venda).		
Atributos			
Nome	Descrição	Domínio	Permite Nulo ? (S/N)
Telefone	Telefone da Concessionária	Texto (10) - ((XX) XXXX-XXXX)	N
CNPJ	Cadastro Nacional da Pessoa Jurídica	Texto (14) - (XX.XXX.XXX/XXXX-XX)	N
CEP	Código de endereçamento postal	Texto (8) - (XXXXX-XXX)	N
Cidade	Nome da cidade	Texto (30)	N
Rua	Rua da concessionária	Texto (30)	N
NumVeiculos	Número de veículos atualmente expostos na concessionária	Inteiro (4) Positivo	S

Tipo Entidade:	Moto		
Descrição:	Um veículo do tipo moto.		
Atributos			
Nome	Descrição	Domínio	Permite Nulo ? (S/N)
Cilindradas	Numero de cilindradas da moto.	Inteiro (3) positivo	N

Tipo Entidade:	Caminhão		
Descrição:	Um veículo do tipo caminhão.		
Atributos			
Nome	Descrição	Domínio	Permite Nulo ? (S/N)
NumEixos	Numero de eixos do caminhão	Inteiro (2) positivo	N
CargaMax	Número que representa quantos KG o caminhao pode carregar	Inteiro (5) positivo	N

Tipo Entidade:	Carro		
Descrição:	Um veículo do tipo carro.		
Atributos			
Nome	Descrição	Domínio	Permite Nulo ? (S/N)
NumPortas	Número de portas do carro	Inteiro (1) positivo	N

Tipo Relacionamento:	Fornece F		
Descrição:	Operação de fornecimento que envolve Fornecedor, indica qual fornecedor está realizando o fornecimento.		

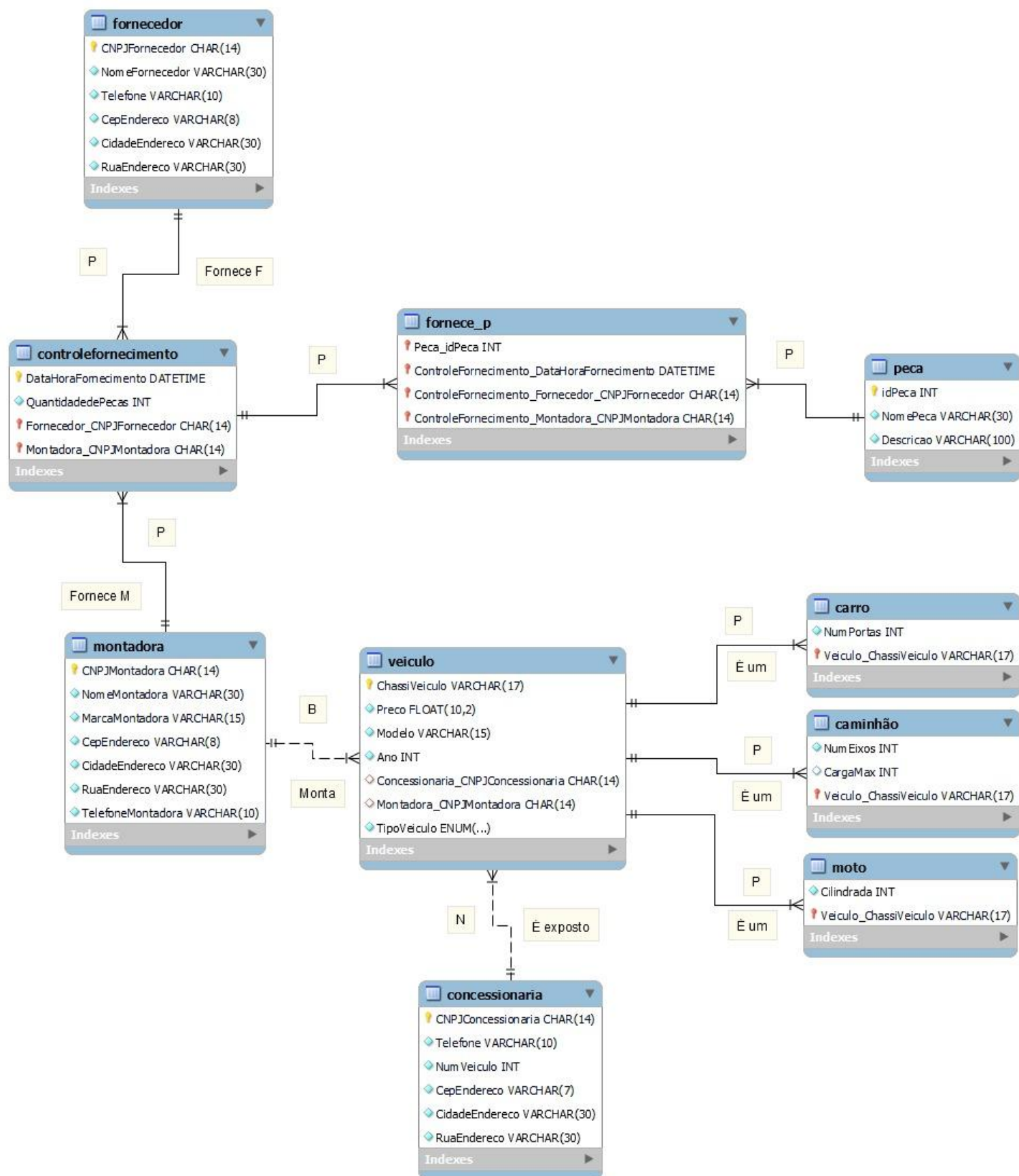
Tipo Relacionamento:	Fornece P		
Descrição:	Operação de fornecimento que envolve Peça, indica quais peças compõem o fornecimento.		

Tipo Relacionamento:	Fornece M		
Descrição:	Operação de fornecimento que envolve Montadora, indica qual montadora irá receber o fornecimento.		

Tipo Relacionamento:	Monta
Descrição:	Representa o processo de montagem de um veículo.

Tipo Relacionamento:	É exposto
Descrição:	Indica a concessionária em que o veículo será exposto.

Diagrama Relacional



Dicionário de dados – Relacional

Tabela	Fornecedor
Descrição	Fornecedores que concedem as peças a montadora
Atributos	Atributos
Nome	Descrição
CNPJFornecedor	Cadastro nacional de pessoa jurídica
NomeFornecedor	Nome do fornecedor
Telefone	Telefone do fornecedor
CepEndereco	Código de endereçamento postal
CidadeEndereco	Nome da cidade
RuaEndereco	Nome da rua

Tabela	Peca
Descrição	Peças fabricadas pelo fornecedor
Atributos	Atributos
Nome	Descrição
IdPeca	Código identificador da peça
NomePeca	Nome da peça
Descricao	Descrição da peça

Tabela	Fornece P
Descrição	Relação entre peça e controle fornecimento
Atributos	Atributos
Nome	Descrição
Peca_idPeca	Referência ao código identificador da peça
Controle FornecimentoDataHoraFornecimento	Referência a data e hora do fornecimento

Tabela	Controle fornecimento
Descrição	Controla a relação do fornecimento de peças entre fornecedor, peça e montadora.
Atributos	Atributos
Nome	Descrição
DataHoraFornecimento	Data e horário em que o fornecimento de peças foi realizado
QuantidadeDePecas	Quantidades de peças fornecidas
Fornecedor_CNPJFornecedor	Referência ao cadastro nacional de pessoa jurídica do fornecedor
Montadora_CNPJMontadora	Referência ao cadastro nacional de pessoa jurídica da montadora

Tabela	Montadora
Descrição	Montadoras de veículos (recebem peças e constroem os veículos para enviar às concessionárias).
Atributos	Atributos
Nome	Descrição
CNPJMontadora	Cadastro Nacional da Pessoa Jurídica
NomeMontadora	Nome da montadora
MarcaMontadora	Marca da montadora
CepEndereco	Código de endereçamento postal
CidadeEndereco	Nome da cidade
RuaEndereco	Rua da montadora

TelefoneMontadora	Telefone da montadora
-------------------	-----------------------

Tabela	Veiculo
Descrição	Conjunto de veículos que são criados
Atributos	Atributos
Nome	Descrição
ChassiVeiculo	Chassi de identificação do veículo

Preco	Valor do veículo
Ano	Ano em que o veículo foi fabricado
Concessionaria_idConcessionaria	Referência ao Código identificador da Concessionária do veículo
Montadora_CNPJMontadora	Referência ao CNPJ da montadora do veículo

Tabela	Concessionaria
Descrição	Unidades de Concessionária de veículos (onde os veículos ficam expostos para venda).
Atributos	Atributos
Nome	Descrição
idConcessionaria	Código identificador da Concessionária
Telefone	Telefone da Concessionária
NumVeiculo	Número de veículos atualmente expostos na concessionária
CepEndereco	Código de endereçamento postal
CidadeEndereco	Nome da cidade
RuaEndereco	Rua da concessionária

Tabela	Moto
Descrição	Um veículo do tipo moto.
Atributos	Atributos
Nome	Descrição
Cilindrada	Número de cilindradas da moto.
Veiculo_ChassiVeiculo	Referência ao Chassi de identificação do veículo

Tabela	Carro
Descrição	Um veículo do tipo carro.
Atributos	Atributos
Nome	Descrição

NumPortas	Número de portas do carro
Veiculo_ChassiVeiculo	Referência ao Chassi de identificação do veículo

Tabela	Caminhão
Descrição	Um veículo do tipo caminhão.
Atributos	Atributos
Nome	Descrição
NumEixos	Número de eixos do caminhão
CargaMax	Número que representa quantos KG o caminhão pode carregar
Veiculo_ChassiVeiculo	Referência ao Chassi de identificação do veículo

Script SQL

```
-- A) CRIAÇÃO
-----

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';

CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 ;
USE `mydb` ;

-----
-- Table `mydb`.`Fornecedor`
-----
CREATE TABLE IF NOT EXISTS `mydb`.`Fornecedor` (
  `CNPJFornecedor` CHAR(14) NOT NULL,
  `NomeFornecedor` VARCHAR(30) NOT NULL,
  `Telefone` VARCHAR(10) NOT NULL,
  `CepEndereco` VARCHAR(8) NOT NULL,
  `CidadeEndereco` VARCHAR(30) NOT NULL,
  `RuaEndereco` VARCHAR(30) NOT NULL,
  UNIQUE INDEX `NomeFornecedor_UNIQUE` (`NomeFornecedor` ASC) VISIBLE,
  PRIMARY KEY (`CNPJFornecedor`))
ENGINE = InnoDB;

-----
-- Table `mydb`.`Montadora`
-----
CREATE TABLE IF NOT EXISTS `mydb`.`Montadora` (
  `CNPJMontadora` CHAR(14) NOT NULL,
  `NomeMontadora` VARCHAR(30) NOT NULL,
  `MarcaMontadora` VARCHAR(15) NOT NULL,
  `CepEndereco` VARCHAR(8) NOT NULL,
  `CidadeEndereco` VARCHAR(30) NOT NULL,
  `RuaEndereco` VARCHAR(30) NOT NULL,
  `TelefoneMontadora` VARCHAR(10) NOT NULL,
  PRIMARY KEY (`CNPJMontadora`),
  UNIQUE INDEX `NomeMontadora_UNIQUE` (`NomeMontadora` ASC) VISIBLE)
ENGINE = InnoDB;
```

```
-- -----
-- Table `mydb`.`ControleFornecimento`
-- -----
CREATE TABLE IF NOT EXISTS `mydb`.`ControleFornecimento` (
  `DataHoraFornecimento` DATETIME NOT NULL,
  `QuantidadeDePecas` INT(6) NOT NULL,
  `Fornecedor_CNPJFornecedor` CHAR(14) NOT NULL,
  `Montadora_CNPJMontadora` CHAR(14) NOT NULL,
  PRIMARY KEY (`DataHoraFornecimento`, `Fornecedor_CNPJFornecedor`, `Montadora_CNPJMontadora`),
  INDEX `fk_Control_Foornecimento_Fornecedor1_idx` (`Fornecedor_CNPJFornecedor` ASC) VISIBLE,
  INDEX `fk_Control_Foornecimento_Montadora1_idx` (`Montadora_CNPJMontadora` ASC) VISIBLE,
  CONSTRAINT `fk_Control_Foornecimento_Fornecedor1`
    FOREIGN KEY (`Fornecedor_CNPJFornecedor`)
      REFERENCES `mydb`.`Fornecedor` (`CNPJFornecedor`)
        ON DELETE RESTRICT
        ON UPDATE NO ACTION,
  CONSTRAINT `fk_Control_Foornecimento_Montadora1`
    FOREIGN KEY (`Montadora_CNPJMontadora`)
      REFERENCES `mydb`.`Montadora` (`CNPJMontadora`)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

```
-- -----
-- Table `mydb`.`Peca`
-- -----
CREATE TABLE IF NOT EXISTS `mydb`.`Peca` (
  `idPeca` INT(3) auto_increment NOT NULL,
  `NomePeca` VARCHAR(30) NOT NULL,
  `Descricao` VARCHAR(100) NOT NULL,
  PRIMARY KEY (`idPeca`))
ENGINE = InnoDB;
```

```
-- -----
-- Table `mydb`.`Concessionaria`
-- -----
CREATE TABLE IF NOT EXISTS `mydb`.`Concessionaria` (
  `CNPJConcessionaria` CHAR(14) NOT NULL,
  `Telefone` VARCHAR(10) NOT NULL,
  `NumVeiculo` INT(4) NOT NULL DEFAULT 0,
  `CepEndereco` VARCHAR(7) NOT NULL,
  `CidadeEndereco` VARCHAR(30) NOT NULL,
  `RuaEndereco` VARCHAR(30) NOT NULL,
  PRIMARY KEY (`CNPJConcessionaria`))
ENGINE = InnoDB;
```

```

-- -----
-- Table `mydb`.`Veiculo`
-- -----
CREATE TABLE IF NOT EXISTS `mydb`.`Veiculo` (
  `ChassiVeiculo` VARCHAR(17) NOT NULL,
  `Preco` FLOAT(10,2) NOT NULL,
  `Modelo` VARCHAR(15) NOT NULL,
  `Ano` INT(4) NOT NULL,
  `Concessionaria_CNPJConcessionaria` CHAR(14) NULL,
  `Montadora_CNPJMontadora` CHAR(14) NULL,
  `TipoVeiculo` ENUM("Moto", "Carro", "Caminhao") NOT NULL,
  PRIMARY KEY (`ChassiVeiculo`),
  INDEX `fk_Veiculo_Concessionaria1_idx` (`Concessionaria_CNPJConcessionaria` ASC) VISIBLE,
  INDEX `fk_Veiculo_Montadora1_idx` (`Montadora_CNPJMontadora` ASC) VISIBLE,
  CONSTRAINT `fk_Veiculo_Concessionaria1`
    FOREIGN KEY (`Concessionaria_CNPJConcessionaria`)
    REFERENCES `mydb`.`Concessionaria` (`CNPJConcessionaria`)
    ON DELETE SET NULL
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Veiculo_Montadora1`
    FOREIGN KEY (`Montadora_CNPJMontadora`)
    REFERENCES `mydb`.`Montadora` (`CNPJMontadora`)
    ON DELETE SET NULL
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-- -----
-- Table `mydb`.`Moto`
-- -----
CREATE TABLE IF NOT EXISTS `mydb`.`Moto` (
  `Cilindrada` INT(3) NOT NULL,
  `Veiculo_ChassiVeiculo` VARCHAR(17) NOT NULL,
  PRIMARY KEY (`Veiculo_ChassiVeiculo`))
ENGINE = InnoDB;

```

```

-- -----
-- Table `mydb`.`Caminhão`
-- -----
CREATE TABLE IF NOT EXISTS `mydb`.`Caminhão` (
  `NumEixos` INT(2) NOT NULL,
  `CargaMax` INT(5) NOT NULL,
  `Veiculo_ChassiVeiculo` VARCHAR(17) NOT NULL,
  PRIMARY KEY (`Veiculo_ChassiVeiculo`),
  CONSTRAINT `fk_Caminhão_Veiculo1`
    FOREIGN KEY (`Veiculo_ChassiVeiculo`)
    REFERENCES `mydb`.`Veiculo` (`ChassiVeiculo`)
    ON DELETE CASCADE
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-- -----
-- Table `mydb`.`Carro`
-- -----
CREATE TABLE IF NOT EXISTS `mydb`.`Carro` (
  `NumPortas` INT(1) NOT NULL,
  `Veiculo_ChassiVeiculo` VARCHAR(17) NOT NULL,
  PRIMARY KEY (`Veiculo_ChassiVeiculo`),
  CONSTRAINT `fk_Carro_Veiculo1`
    FOREIGN KEY (`Veiculo_ChassiVeiculo`)
    REFERENCES `mydb`.`Veiculo` (`ChassiVeiculo`)
    ON DELETE CASCADE
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

-- -----
-- Table `mydb`.`Fornece_P`
-- -----
CREATE TABLE IF NOT EXISTS `mydb`.`Fornece_P` (
  `Peca_idPeca` INT(3) NOT NULL,
  `ControleFornecimento_DataHoraFornecimento` DATETIME NOT NULL,
  `ControleFornecimento_Fornecedor_CNPJFornecedor` CHAR(14) NOT NULL,
  `ControleFornecimento_Montadora_CNPJMontadora` CHAR(14) NOT NULL,
  PRIMARY KEY (`Peca_idPeca`, `ControleFornecimento_DataHoraFornecimento`,
`ControleFornecimento_Fornecedor_CNPJFornecedor`, `ControleFornecimento_Montadora_CNPJMontadora`),
  INDEX `fk_Peca_has_ControlFornecimento_ControlFornecimento1_idx`
(`ControleFornecimento_DataHoraFornecimento` ASC, `ControleFornecimento_Fornecedor_CNPJFornecedor` ASC,
`ControleFornecimento_Montadora_CNPJMontadora` ASC) VISIBLE,
  INDEX `fk_Peca_has_ControlFornecimento_Peca1_idx` (`Peca_idPeca` ASC) VISIBLE,
  CONSTRAINT `fk_Peca_has_ControlFornecimento_Peca1`
    FOREIGN KEY (`Peca_idPeca`)
    REFERENCES `mydb`.`Peca` (`idPeca`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Peca_has_ControlFornecimento_ControlFornecimento1`
    FOREIGN KEY (`ControleFornecimento_DataHoraFornecimento`,
`ControleFornecimento_Fornecedor_CNPJFornecedor`, `ControleFornecimento_Montadora_CNPJMontadora`)
    REFERENCES `mydb`.`ControleFornecimento` (`DataHoraFornecimento`, `Fornecedor_CNPJFornecedor`,
`Montadora_CNPJMontadora`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

```



```

-- B) ALTERAÇÕES E DROPS
-----

-- Valor DEFAULT de 0 para 1
ALTER TABLE Concessionaria ALTER COLUMN NumVeiculo SET DEFAULT 1;

-- Adicionar a Constraint da chave estrangeira para a tabela MOTO (que foi criada sem FK)
ALTER TABLE moto ADD
CONSTRAINT `fk_Moto_Veiculo1`
    FOREIGN KEY (`Veiculo_ChassiVeiculo`)
    REFERENCES `mydb`.`Veiculo` (`ChassiVeiculo`)
    ON DELETE CASCADE
    ON UPDATE NO ACTION;

-- select moto.cilindrada, moto.veiculo_chassiveiculo from moto natural join veiculo;

-- INT de tamanho 5 para 10
ALTER TABLE Caminhão MODIFY COLUMN CargaMax INT(10);

-- -----
-- Table `mydb`.`Ônibus`
-- -----

CREATE TABLE IF NOT EXISTS `mydb`.`Ônibus` (
    `NumPoltronas` INT(1) NOT NULL,
    `Veiculo_ChassiVeiculo` VARCHAR(17) NOT NULL,
    PRIMARY KEY (`Veiculo_ChassiVeiculo`),
    CONSTRAINT `fk_Onibus_Veiculo1`
        FOREIGN KEY (`Veiculo_ChassiVeiculo`)
        REFERENCES `mydb`.`Veiculo` (`ChassiVeiculo`)
        ON DELETE CASCADE
        ON UPDATE NO ACTION)
ENGINE = InnoDB;

-- Exclusão da tabela Ônibus
DROP TABLE Ônibus;

```

-- C) INSERÇÕES

```
INSERT INTO Fornecedor (CNPJFornecedor, NomeFornecedor, Telefone, CepEndereco, CidadeEndereco, RuaEndereco) VALUES
('000000000000000', 'TRW', '1111111111', '02860001', 'São Paulo', 'Avenida Deputado Cantídio'),
('111111111111111', 'Positrom', '2222222222', '03018000', 'São Paulo', 'Rua João Boemer'),
('222222222222222', 'Lubrax', '3333333333', '59114250', 'Natal', 'Avenida Boa Sorte'),
('333333333333333', 'Goodyear', '4444444444', '35500008', 'Divinópolis', 'Rua Pernambuco'),
('444444444444444', 'Suspensys', '5555555555', '57061110', 'Maceió', 'Avenida Maceió'),
('555555555555555', 'Tecfill', '6666666666', '91170200', 'Porto Alegre', 'Rua dos Maias'),
('666666666666666', 'Arteb', '7777777777', '31720300', 'Belo Horizonte', 'Avenida Doutor Cristiano'),
('777777777777777', 'Nakata', '8888888888', '69075771', 'Manaus', 'Avenida Rio Negro'),
('888888888888888', 'Taramp´s', '9999999999', '04428000', 'São Paulo', 'Avenida Yervant Kissajikian'),
('999999999999999', 'Pirelli', '1010101010', '38408168', 'Uberlândia', 'Avenida Belarmino Cotta');
```

```
INSERT INTO Peca ( NomePeca, Descricao ) VALUES
( 'Pneu', 'pneu 165/60 carro'),
( 'Pneu', 'pneu 175/70 carro'),
( 'Alto-falante', 'Alto falante 130 watts'),
( 'Subwoofer', 'subwoofer 15 polegadas'),
( 'Amortecedor Carro', 'Amortecedor carro dianteiro'),
( 'Amortecedor Carro', 'Amortecedor traseiro'),
( 'Farol', 'farol dianteiro carro'),
( 'Lanterna', 'Lanterna traseira moto'),
( 'Lanterna', 'Lanterna Traseira carro'),
( 'Pneu', 'pneu 60/100-17 moto');
```

```
INSERT INTO Montadora (CNPJMontadora, NomeMontadora, MarcaMontadora, CepEndereco, CidadeEndereco, RuaEndereco, TelefoneMontadora) VALUES
('000000000000001', 'Toyota', 'Toyota Motor', '09710010', 'São Bernardo do Campo', 'Rua Marechal Deodoro', '1111111112'),
('111111111111112', 'Lexus', 'Toyota Motor', '09761000', 'São Bernardo do Campo', 'Rua dos Vianas', '2222222223'),
('222222222222223', 'Honda', 'Honda', '13010110', 'Campinas', 'Rua General Osório', '3333333334'),
('333333333333334', 'Fiat', 'Stellantis', '03063000', 'São Paulo', 'Avenida Celso Garcia', '4444444445'),
('444444444444445', 'Jeep', 'Stellantis', '01130000', 'São Paulo', 'Rua Anhaia', '5555555556'),
('555555555555556', 'Peugeot', 'Stellantis', '03023000', 'São Paulo', 'Rua Rio Bonito', '6666666667'),
('666666666666667', 'Ferrari', 'Stellantis', '86010190', 'Londrina', 'Avenida Duque de Caxias', '7777777778'),
('777777777777778', 'Volkswagen', 'GrupoVolkswagen', '29045300', 'Vitória', 'Rua José Farias', '8888888889'),
('888888888888889', 'Audi', 'GrupoVolkswagen', '50090000', 'Recife', 'Rua Imperial', '9999999990'),
('999999999999990', 'Acura', 'Honda', '40470630', 'Salvador', 'Avenida Afrânio Peixoto', '1010101011');
```

```
INSERT INTO ControleFornecimento (DataHoraFornecimento, QuantidadedePecas, Fornecedor_CNPJFornecedor,
Montadora_CNPJMontadora) VALUES
('2023-02-20', '250', '00000000000000', '00000000000001'),
('2023-02-19', '396', '66666666666666', '44444444444445'),
('2023-02-18', '1024', '66666666666666', '33333333333334'),
('2023-02-18', '212', '88888888888888', '77777777777778'),
('2023-02-20', '540', '88888888888888', '33333333333334'),
('2023-02-19', '60', '88888888888888', '55555555555556'),
('2023-02-15', '44', '99999999999999', '66666666666667'),
('2023-02-10', '664', '11111111111111', '44444444444445'),
('2023-02-20', '700', '11111111111111', '22222222222223'),
('2023-02-13', '486', '00000000000000', '11111111111112');
```

```
INSERT INTO Fornece_P (Peca_idPeca, ControleFornecimento_DataHoraFornecimento,
ControleFornecimento_Fornecedor_CNPJFornecedor, ControleFornecimento_Montadora_CNPJMontadora) VALUES
('3', '2023-02-10 00:00:00', '11111111111111', '44444444444445'),
('4', '2023-02-10 00:00:00', '11111111111111', '22222222222223'),
('5', '2023-02-13 00:00:00', '00000000000000', '11111111111112'),
('5', '2023-02-20 00:00:00', '00000000000000', '00000000000001'),
('2', '2023-02-15 00:00:00', '99999999999999', '66666666666667'),
('5', '2023-02-18 00:00:00', '66666666666666', '33333333333334'),
('6', '2023-02-19 00:00:00', '66666666666666', '44444444444445'),
('4', '2023-02-18 00:00:00', '88888888888888', '77777777777778'),
('3', '2023-02-19 00:00:00', '88888888888888', '55555555555556'),
('3', '2023-02-20 00:00:00', '88888888888888', '33333333333334');
```

```
INSERT INTO Concessionaria (CNPJConcessionaria, Telefone, NumVeiculo, CepEndereco, CidadeEndereco,
RuaEndereco) VALUES
('85000009000001', '3560001000', 570, '3517000', 'Belo Horizonte', 'Cubatão'),
('76000008000012', '3560001011', 230, '3527001', 'Ipatinga', 'Ibiritatí'),
('67000007000023', '3560001022', 300, '3537002', 'Vitória', 'Catanduva'),
('58000006000034', '3560001033', 300, '3547003', 'Lavras', 'Gravatá'),
('49000005000045', '3560001044', 480, '3557004', 'Viçosa', 'Jacuí'),
('37000004000056', '3560001055', 500, '3567005', 'Juiz de Fora', 'Salgueiro'),
('28000003000067', '3560001066', 520, '3577006', 'Barbacena', 'São Roque'),
('19000002000078', '3560001077', 170, '3587007', 'Varginha', 'Horizontal'),
('94000001000089', '3560001088', 350, '3597008', 'Uberlândia', 'Pitangui'),
('07200000000090', '3560001099', 700, '3507009', 'Belo Horizonte', 'Outono');
```

```
INSERT INTO Veiculo (ChassiVeiculo, Preco, Modelo, Ano, Concessionaria_CNPJConcessionaria,
Montadora_CNPJMontadora, TipoVeiculo) VALUES
('9BG116GW04C900001',35000.99,'Gol',2019,'85000009000001','77777777777778','Carro'),
('9BG116GW04C700002',39000.99,'Uno',2017,'76000008000012','33333333333334','Carro'),
('9BG116GW04C100003',200000.99,'Compass',2021,'76000008000012','44444444444445','Carro'),
('9BG116GW04C200004',999000.99,'Roma',2023,'67000007000023','66666666666667','Carro'),
('9BG116GW04C900005',45000.99,'Etios',2019,'67000007000023','00000000000001','Carro'),
('9BG116GW04C500006',140000.99,'Renegade',2021,'58000006000034','44444444444445','Carro'),
('9BG116GW04C900011',8000.99,'Speedfight',2015,'85000009000001','55555555555556','Moto'),
('9BG116GW04C700012',9000.99,'C 100 BIZ',2017,'76000008000012','22222222222223','Moto'),
('9BG116GW04C100013',10000.99,'125',2021,'37000004000056','22222222222223','Moto'),
('9BG116GW04C200014',25000.99,'750 MAGNA',2023,'28000003000067','22222222222223','Moto'),
('9BG116GW04C900015',9000.99,'ADV',2019,'67000007000023','22222222222223','Moto'),
('9BG116GW04C500016',20500.99,'1200 GOLD WING',2021,'37000004000056','22222222222223','Moto'),
('9BG116GW04C900021',570725.99,'FH 460',2019,'19000002000078','11111111111112','Caminhão'),
('9BG116GW04C700022',714285.99,'Actros 2651',2017,'19000002000078','11111111111112','Caminhão'),
('9BG116GW04C100023',629200.99,'R540',2021,'94000001000089','88888888888889','Caminhão'),
('9BG116GW04C200024',595306.99,'R500',2023,'94000001000089','88888888888889','Caminhão'),
('9BG116GW04C900025',516667.99,'Axor 2544',2019,'07200000000090','99999999999990','Caminhão'),
('9BG116GW04C500026',660400.99,'Meteor 29.520',2021,'07200000000090','77777777777778','Caminhão');
```

```
INSERT INTO Carro (NumPortas, Veiculo_ChassiVeiculo) VALUES
(4,'9BG116GW04C900001'),
(4,'9BG116GW04C700002'),
(4,'9BG116GW04C100003'),
(4,'9BG116GW04C200004'),
(4,'9BG116GW04C900005'),
(4,'9BG116GW04C500006');
```

```
INSERT INTO Moto (Cilindrada, Veiculo_ChassiVeiculo) VALUES
(750, '9BG116GW04C900011'),
(620, '9BG116GW04C700012'),
(600, '9BG116GW04C100013'),
(739, '9BG116GW04C200014'),
(690, '9BG116GW04C900015'),
(890, '9BG116GW04C500016');
```

```
INSERT INTO Caminhão (NumEixos, CargaMax, Veiculo_ChassiVeiculo) VALUES
(9, 99999, '9BG116GW04C900021'),
(8, 99999, '9BG116GW04C700022'),
(10, 99999, '9BG116GW04C100023'),
(8, 99999, '9BG116GW04C200024'),
(9, 99999, '9BG116GW04C900025'),
(10, 99999, '9BG116GW04C500026');
```

```
-- SET SQL_MODE=@OLD_SQL_MODE;
-- SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
-- SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

-- D) UPDATES

/*

Alterar o endereço da montadora de Nome ferrari para
Cep 81720000, Cidade Curitiba, Rua Rua Francisco Derosso.

*/

UPDATE Montadora

SET CepEndereco = 81720000, CidadeEndereco = 'Curitiba', RuaEndereco = 'Rua Francisco Derosso'
WHERE NomeMontadora = 'Ferrari';

/*

Alterar o endereço do Fornecedor de Nome Goodyear para
Cep 37500016, Cidade Itajubá, Rua Major Belo Lisboa.

*/

UPDATE Fornecedor

SET CepEndereco = 37500016, CidadeEndereco = 'Itajubá', RuaEndereco = 'Rua Major Belo Lisboa'
WHERE NomeFornecedor = 'Goodyear';

/*

Alterar a descrição da peça de id 3 para
'Alto falante 100 watts JBL'

*/

UPDATE Peca

SET Descricao = 'Alto falante 100 watts JBL'
WHERE idPeca = 3;

/*

Alterar a quantidade de veículos no estoque da concessionária de
cnpj = 19000002000078 para 200.

*/

UPDATE Concessionaria

SET NumVeiculo = 200
WHERE CNPJConcessionaria = '19000002000078';

/*

Alterar o preço e modelo do carro de chassi 9BG116GW04C500006 para
modelo = Renegade, preço = 145999,99.

*/

UPDATE Veiculo

SET Modelo = 'Renegade', Preço =145999.99
WHERE ChassiVeiculo = '9BG116GW04C500006';

/*

UPDATE aninhado envolvendo as tabelas veiculo e moto, em que o preço das motos
de cilindrada superior a 700 vão aumentar em 10%.

*/

UPDATE Veiculo

SET Preço = Preço*1.1
WHERE TipoVeiculo = 'Moto' and ChassiVeiculo IN (SELECT Veiculo_ChassiVeiculo
FROM Moto
WHERE Cilindrada > 700);

```

-- E) DELETES
-----

-- Exclui a montadora Acura
DELETE FROM Montadora
WHERE NomeMontadora = 'Acura';

-- Exclui o Fornecedor Nakata
DELETE FROM Fornecedor
WHERE NomeFornecedor = 'Nakata';

-- Exclui a peça de id 1
DELETE FROM Peca
WHERE idPeca = 1;

-- Exclui a concessionária de cnpj 28000003000067
DELETE FROM Concessionaria
WHERE CNPJConcessionaria = 28000003000067;

/*
DELETE aninhado envolvendo as tabelas veiculo e caminhão, em que exclui um veículo
que o número de eixos é igual a 8 e o tipo do veículo é caminhão.
*/
DELETE FROM Veiculo
WHERE TipoVeiculo = 'Caminhao' and ChassiVeiculo IN (SELECT Veiculo_ChassiVeiculo
                                                    FROM Caminhão
                                                    WHERE NumEixos = 8);

-- F) CONSULTAS
-----

-- 1) Seleciona todas as montadoras

SELECT * FROM Montadora;

-- 2) Seleciona o nome de todas as peças da tabela Peca

SELECT NomePeca FROM Peca;

-- 3) Seleciona a quantidade de pecas na tabela Peca

SELECT COUNT(*) AS qtd_pecas FROM Peca;

-- 4) Seleciona o nome de todas as peças em que contem a sentença "pneu" em sua Descricao.

SELECT nomepECA FROM Peca WHERE Descricao LIKE "%pneu%";

```

-- 5) Seleciona o nome e o telefone de todos os fornecedores em que a cidade de endereço é Belo Horizonte.

```
SELECT NomeFornecedor, Telefone
FROM Fornecedor
WHERE CidadeEndereco = 'Belo Horizonte';
```

-- 6) Essa consulta faz uma contagem do número de peças fornecidas por cada montadora. Para isso, ela faz um JOIN entre as tabelas Montadora, ControleFornecimento e Fornece_P, relacionando-as através das chaves primárias e estrangeiras correspondentes. Em seguida, ela agrupa os resultados pelo nome de cada montadora e utiliza a função COUNT(*) para contar o número de registros encontrados para cada montadora. O resultado final apresenta uma tabela com duas colunas: a primeira contém o nome de cada montadora e a segunda contém a quantidade total de peças fornecidas por cada uma delas.

```
SELECT Montadora.NomeMontadora, COUNT(*) AS QuantidadePecas
FROM Montadora
JOIN ControleFornecimento ON Montadora.CNPJMontadora = ControleFornecimento.Montadora_CNPJMontadora
JOIN Fornece_P ON ControleFornecimento.DataHoraFornecimento =
Fornece_P.ControleFornecimento_DataHoraFornecimento
    AND ControleFornecimento.Fornecedor_CNPJFornecedor =
Fornece_P.ControleFornecimento_Fornecedor_CNPJFornecedor
    AND ControleFornecimento.Montadora_CNPJMontadora =
Fornece_P.ControleFornecimento_Montadora_CNPJMontadora
GROUP BY Montadora.NomeMontadora;
```

-- 7) Essa consulta retorna as informações das peças que são fornecidas por fornecedores diferentes de um fornecedor específico com CNPJ '88888888888888' ou que não são fornecidas por nenhum fornecedor. A consulta faz uma junção (JOIN) entre as tabelas "Peca" e "Fornece_P" utilizando a cláusula LEFT JOIN. Isso significa que a consulta retorna todas as linhas da tabela "Peca", mesmo que não haja correspondência na tabela "Fornece_P".

```
SELECT Peca.idPeca, Peca.NomePeca, Fornece_P.ControleFornecimento_Fornecedor_CNPJFornecedor
FROM Peca
LEFT JOIN Fornece_P ON Peca.idPeca = Fornece_P.Peca_idPeca
WHERE Fornece_P.ControleFornecimento_Fornecedor_CNPJFornecedor <> '88888888888888'
    OR Fornece_P.ControleFornecimento_Fornecedor_CNPJFornecedor IS NULL;
```

-- 8) Essa consulta retorna uma lista de fornecedores e a quantidade de peças que eles forneceram, ordenados em ordem decrescente de quantidade de peças. A consulta começa com a junção das tabelas ControleFornecimento e Fornecedor usando a chave estrangeira Fornecedor_CNPJFornecedor da tabela ControleFornecimento e a chave primária CNPJFornecedor da tabela Fornecedor. Em seguida, é utilizado o comando GROUP BY para agrupar as linhas por Fornecedor_CNPJFornecedor e NomeFornecedor, e é utilizado a função COUNT(*) para contar a quantidade de linhas (ou seja, de peças fornecidas) para cada grupo. Por fim, a lista resultante é ordenada em ordem decrescente de quantidade de peças usando o comando ORDER BY quantidade_pecas DESC.

```
SELECT ControleFornecimento.Fornecedor_CNPJFornecedor, Fornecedor.NomeFornecedor, COUNT(*) as
quantidade_pecas
FROM ControleFornecimento
JOIN Fornecedor ON ControleFornecimento.Fornecedor_CNPJFornecedor = Fornecedor.CNPJFornecedor
GROUP BY ControleFornecimento.Fornecedor_CNPJFornecedor, Fornecedor.NomeFornecedor
ORDER BY quantidade_pecas DESC;
```

-- 9) Essa consulta retorna o nome e a cidade da montadora que fornece peças com ID 1, 2, 3, 7, 8 ou 9 para fornecedores cujo CNPJ contenha "99" ou "88". A consulta realiza diversas junções entre as tabelas Montadora, ControleFornecimento, Fornecedor e Fornece_P para obter as informações necessárias. Depois, filtra os resultados para incluir apenas os fornecedores que fornecem peças com os IDs especificados e cujo CNPJ contém "99" ou "88". Por fim, agrupa os resultados por nome de montadora e cidade, ordenando pelo nome da montadora.

```
SELECT M.NomeMontadora, M.CidadeEndereco
FROM Montadora AS M
JOIN ControleFornecimento AS CF ON CF.Montadora_CNPJMontadora = M.CNPJMontadora
JOIN Fornecedor AS F ON F.CNPJFornecedor = CF.Fornecedor_CNPJFornecedor
JOIN Fornece_P AS FP ON FP.ControleFornecimento_DataHoraFornecimento = CF.DataHoraFornecimento
                        AND FP.ControleFornecimento_Fornecedor_CNPJFornecedor = CF.Fornecedor_CNPJFornecedor
                        AND FP.ControleFornecimento_Montadora_CNPJMontadora = CF.Montadora_CNPJMontadora
WHERE FP.Peca_idPeca IN (1,2,3,7,8,9)
      AND (F.CNPJFornecedor LIKE '%99%' OR F.CNPJFornecedor LIKE '%88%')
GROUP BY M.NomeMontadora, M.CidadeEndereco
ORDER BY M.NomeMontadora;
```

-- 10) Essa consulta retorna o nome de todas as montadoras e a quantidade de peças que cada uma fornece. As informações são obtidas a partir de uma junção entre as tabelas Montadora e Fornece_P. A cláusula GROUP BY é usada para agrupar os resultados por montadora e a função de agregação COUNT(*) é usada para contar a quantidade de peças fornecidas por cada montadora. A cláusula HAVING é usada para filtrar os resultados e retornar apenas as montadoras que fornecem mais de uma peça.

```
SELECT Montadora.NomeMontadora, COUNT(*) as qtde_pecas
FROM Montadora
JOIN Fornece_P ON Montadora.CNPJMontadora = Fornece_P.ControleFornecimento_Montadora_CNPJMontadora
GROUP BY Montadora.nomeMontadora
HAVING COUNT(*) > 1;
```

-- 11) Essa consulta seleciona o modelo de todos os caminhões que têm mais eixos do que todos os caminhões que têm exatamente 9 eixos. A cláusula FROM especifica que estamos unindo duas tabelas: Veiculo e Caminhão. A cláusula JOIN especifica que estamos unindo as tabelas através da coluna ChassiVeiculo na tabela Veiculo e da coluna Veiculo_ChassiVeiculo na tabela Caminhão. A cláusula WHERE filtra os resultados para retornar apenas os caminhões que têm mais eixos do que todos os caminhões que têm exatamente 9 eixos. Para fazer isso, usamos a função ALL para comparar o valor da coluna NumEixos de cada caminhão com o valor da subconsulta que retorna todos os caminhões que têm 9 eixos. Finalmente, especificamos quais colunas queremos que sejam retornadas na consulta: o modelo do veículo na tabela Veiculo e o número de eixos do caminhão na tabela Caminhão.

```
SELECT Veiculo.Modelo, Caminhão.NumEixos from Veiculo join Caminhão on Veiculo.ChassiVeiculo =
Caminhão.Veiculo_ChassiVeiculo where Caminhão.NumEixos >ALL (SELECT Veiculo_ChassiVeiculo FROM Caminhão
WHERE NumEixos=9);
```

-- 12) Essa consulta SQL usa a cláusula UNION para combinar duas consultas diferentes em um único resultado e retorna todas as informações sobre todos os veículos que são caminhões e têm um preço entre 500.000 e 1.000.000 ou têm um preço entre 100.000 e 500.000.

```
SELECT * from Veiculo WHERE TipoVeiculo='Caminhão' AND Preço BETWEEN 500000 AND 1000000 UNION SELECT *
FROM Veiculo WHERE Preço BETWEEN 100000 AND 500000;
```


-- 13) Essa consulta retorna os modelos e anos dos veículos cujos preços são menores que pelo menos um preço de outro veículo que pertence a uma montadora específica. Mais especificamente, a subconsulta na cláusula WHERE é responsável por encontrar pelo menos um veículo que pertence à montadora com o CNPJ "2222222222223". Em seguida, a cláusula SOME é usada para comparar o preço dos veículos da consulta principal com esse preço encontrado na subconsulta e retorna verdadeiro se pelo menos um valor na subconsulta satisfaz a comparação especificada.

```
SELECT Modelo, Ano
FROM Veiculo
WHERE Preco < SOME (SELECT Preco from Veiculo WHERE EXISTS (
    SELECT * FROM Montadora WHERE CNPJMontadora = 2222222222223
) AND Montadora_CNPJMontadora = 2222222222223);
```

-- G) VIEWS

-- VIEW que mostra todos os carros e seu ano

```
CREATE VIEW ModelosDeCarros AS
    SELECT Modelo, Ano
    FROM Veiculo
    WHERE TipoVeiculo = 'Carro';
```

```
SELECT * FROM ModelosDeCarros;
```

-- VIEW que mostra o telefone e o número de veículos de todas concessionárias de BH

```
CREATE VIEW ConcessionariaDeBH AS
    SELECT CidadeEndereco, Telefone, NumVeiculo
    FROM Concessionaria
    WHERE CidadeEndereco = 'Belo Horizonte';
```

```
SELECT * FROM ConcessionariaDeBH;
```

-- VIEW que mostra todos os fornecedores de auto-falante

```
CREATE VIEW FornecedorAutoFalante AS
    SELECT DISTINCT NomeFornecedor
    FROM Fornecedor F, Peca P, ControleFornecimento C, Fornece_P FP
    WHERE P.NomePeca = 'Alto-falante' AND F.CNPJFornecedor = C.Fornecedor_CNPJFornecedor AND C.
Fornecedor_CNPJFornecedor = FP.ControleFornecimento_Fornecedor_CNPJFornecedor AND FP.Peca_idPeca =
P.idPeca;
```

```
SELECT * FROM FornecedorAutoFalante;
```

```
-- H) USUÁRIOS
```

```
-----  
-----  
  
-- Cria os usuarios
```

```
CREATE USER 'wesley'@'localhost' identified by "senha123";  
CREATE USER 'ronald'@'localhost' identified by "senha321";
```

```
-- Dá permissão de super usuário ao 'wesley'
```

```
GRANT ALL ON mydb.* TO 'wesley'@'localhost';
```

```
-- Usuario que pode somente realizar SELECT'S na tabela "Fornecedor"
```

```
grant select on mydb.fornecedor to 'ronald'@'localhost';
```

```
-- Remove a permissão de SELECT do usuário
```

```
REVOKE select ON mydb.fornecedor FROM 'ronald'@'localhost';
```

```
-- Remove todas as permissões do usuário
```

```
revoke all on mydb.* FROM 'wesley'@'localhost';
```

```
drop user 'wesley'@'localhost';
```

```
drop user 'ronald'@'localhost';
```

```
-- I) STORED PROCEDURES
```

```
-----  
-----  
  
-- 1º - Procurar veículos de acordo com um Chassi inserido
```

```
delimiter //
```

```
create procedure procurarVeiculo(in pChassi VARCHAR(17) )
```

```
begin
```

```
    select * from veiculo where ChassiVeiculo = pChassi;
```

```
end //
```

```
delimiter ;
```

```
-- 2º - Contar e Exibir a quantidade de fornecedores cadastrados no Banco de Dados da empresa e exibir uma  
mensagem referente a quantidade
```

```
delimiter //
```

```
create procedure contarFornecedores(out pNumFornecedores int , out pMessage varchar (30))
```

```
begin
```

```
    select count(*) into pNumFornecedores from fornecedor;
```

```
    if pNumFornecedores < 7 then set pMessage = "Poucos Fornecedores";
```

```
    elseif pNumFornecedores >= 7 and pNumFornecedores <= 15 then set pMessage = "Alguns Fornecedores";
```

```
    else set pMessage = "Muitos Fornecedores";
```

```
    end if;
```

```
end //
```

```
delimiter ;
```

```

-- 3º - Aumentar em 10% o preço dos carros que tem ano > 2020
delimiter //
create procedure ajuste()
begin
    declare done int default false;
    declare vChassiVeiculo varchar(17);
    declare vPreco float(10,2);
    declare vAno int (4);
    declare curs cursor for
        select ChassiVeiculo, Preco, Ano
        from veiculo;
    declare continue handler for not found set done = true;
    open curs;
    read_loop: loop
        fetch curs into vChassiVeiculo, vPreco, vAno;
        if done then
            leave read_loop;
        end if;
        if vAno > 2020 then
            update Veiculo set preco = preco * 1.1 where ChassiVeiculo = vChassiVeiculo;
        end if;
    end loop;
    close curs;
end //
delimiter ;

-- 4º - Exibir o Endereço simplificado de uma montadora inserida ( Vai alterar o atributo de Endereço para
escrever "R." e "Av." ao invés de "Rua" ou "Avenida")
delimiter //
create procedure simplificarLocal(in pInput varchar(100))
begin
    declare var varchar(100);
    select RuaEndereco into var from montadora where NomeMontadora = pInput;
    if var like "%rua%" then
        UPDATE mydb.montadora
        SET RuaEndereco = CONCAT('R. ', SUBSTRING(RuaEndereco, 3))
        WHERE montadora.NomeMontadora = pInput;
    elseif var like "%avenida%" then
        UPDATE mydb.montadora
        SET RuaEndereco = CONCAT('Av. ', SUBSTRING(RuaEndereco, 9))
        WHERE montadora.NomeMontadora = pInput;
    end if;
    select NomeMontadora, CidadeEndereco, RuaEndereco from montadora where
montadora.NomeMontadora = pInput;
end //
delimiter ;

```

```

-- CHAMAR 1º
call procurarVeiculo("9BG116GW04C200004");

-- CHAMAR 2º
INSERT INTO Fornecedor (CNPJFornecedor, NomeFornecedor, Telefone, CepEndereco, CidadeEndereco,
RuaEndereco) VALUES
('999299999999999', 'Pir', '1010101010', '38408168', 'Uberlândia', 'Avenida Belarmino Cotta'),
('888388888888888', 'Taram', '9999999999', '04428000', 'São Paulo', 'Avenida Yervant Kissajikian'),
('888488888888888', 'Taraxx', '9999999999', '04428000', 'São Paulo', 'Avenida Yervant Kissajikian'),
('888588888888888', 'amp´s', '9999999999', '04428000', 'São Paulo', 'Avenida Yervant Kissajikian'),
('888688888888888', 'mp´s', '9999999999', '04428000', 'São Paulo', 'Avenida Yervant Kissajikian'),
('888788888888888', 'ram', '9999999999', '04428000', 'São Paulo', 'Avenida Yervant Kissajikian');
call contarFornecedores(@numFornecedores, @message);
select @numFornecedores as Numero_Fornecedores, @message as Classificacao;

-- CHAMAR 3º
select * from veiculo;
call ajuste();
select * from veiculo;

-- CHAMAR 4º
INSERT INTO Montadora (CNPJMontadora, NomeMontadora, MarcaMontadora, CepEndereco, CidadeEndereco,
RuaEndereco, TelefoneMontadora) VALUES
('00043000895001', 'test1', 'Toyota Motor', '09710010', 'São Bernardo do Campo', 'Avenida Teste',
'1111111112');
INSERT INTO Montadora (CNPJMontadora, NomeMontadora, MarcaMontadora, CepEndereco, CidadeEndereco,
RuaEndereco, TelefoneMontadora) VALUES
('01043000895001', 'test2', 'Toyota Motor', '09710010', 'São Bernardo do Campo', 'Rua Teste',
'1111111112');
INSERT INTO Montadora (CNPJMontadora, NomeMontadora, MarcaMontadora, CepEndereco, CidadeEndereco,
RuaEndereco, TelefoneMontadora) VALUES
('01043006895001', 'test3', 'Toyota Motor', '09710010', 'São Bernardo do Campo', 'Alameda Teste',
'1111111112');
call simplificarLocal("test1");
call simplificarLocal("test2");
call simplificarLocal("test3");

-- DROPAR
drop procedure procurarVeiculo;
drop procedure contarFornecedores;
drop procedure ajuste;
drop procedure simplificarLocal;

```

```
-- J) TRIGGERS
-----
-----

-- Criação de TRIGGERS de INSERT, UPDATE e DELETE que inserem dados em uma tabela de HISTORICO referentes
à alterações feitas na tabela de PECA

CREATE TABLE IF NOT EXISTS `mydb`.`historicoPeca` (
  Operacao varchar(10),
  idPeca INT(3) NOT NULL,
  NomePeca VARCHAR(30) NOT NULL,
  Descricao VARCHAR(100) NOT NULL,
  Data DATETIME);

-- TRIGGER DE INSERT
delimiter //
create trigger trigger_Insert_Peca
after insert on peca
for each row
begin
    insert into historicoPeca(Operacao, idPeca, NomePeca, Descricao, Data) values ("INSERT",
NEW.idPeca, NEW.NomePeca, NEW.Descricao, now());
end //
delimiter ;

-- TRIGGER DE UPDATE
delimiter //
create trigger trigger_Update_Peca
after UPDATE on peca
for each row
begin
    insert into historicoPeca(Operacao, idPeca, NomePeca, Descricao, Data) values ("UPDATE",
NEW.idPeca, NEW.NomePeca, NEW.Descricao, now());
end //
delimiter ;

-- TRIGGER DE DELETE
delimiter //
create trigger trigger_Delete_Peca
after DELETE on peca
for each row
begin
    insert into historicoPeca(Operacao, idPeca, NomePeca, Descricao, Data) values ("DELETE",
OLD.idPeca, OLD.NomePeca, OLD.Descricao, now());
end //
delimiter ;

-- Ativação
insert into mydb.pecas(NomePeca, Descricao) values ("Teste_Trigger", "muito legal");
UPDATE Peca SET Descricao = 'Sofri Update em' WHERE NomePeca = "Teste_Trigger";
DELETE FROM Peca WHERE NomePeca = "Teste_Trigger";

-- Delete dos TRIGGERS
drop trigger trigger_Insert_Peca;
drop trigger trigger_Delete_Peca;
drop trigger trigger_Update_Peca;
```

```
drop table historicoPeca;
```

Link para o Código SQL (se necessário para melhor observação):

https://drive.google.com/drive/folders/1nYOi90Ow-j2bu3t3Qj4j1Wz9FB2-Neg0?usp=share_link