

# Lista exercícios “Criação Tabelas” - SSMS

1.

## - Tabela 1 – Pet:

```
create database ExercBD01 --criando banco
```

```
use ExercBD01
```

```
CREATE TABLE pet
```

```
(
```

```
NumRegistro int, --criando campo registro
```

```
Nome varchar(80), --criando campo nome
```

```
Especie varchar(25), --criando campo especie
```

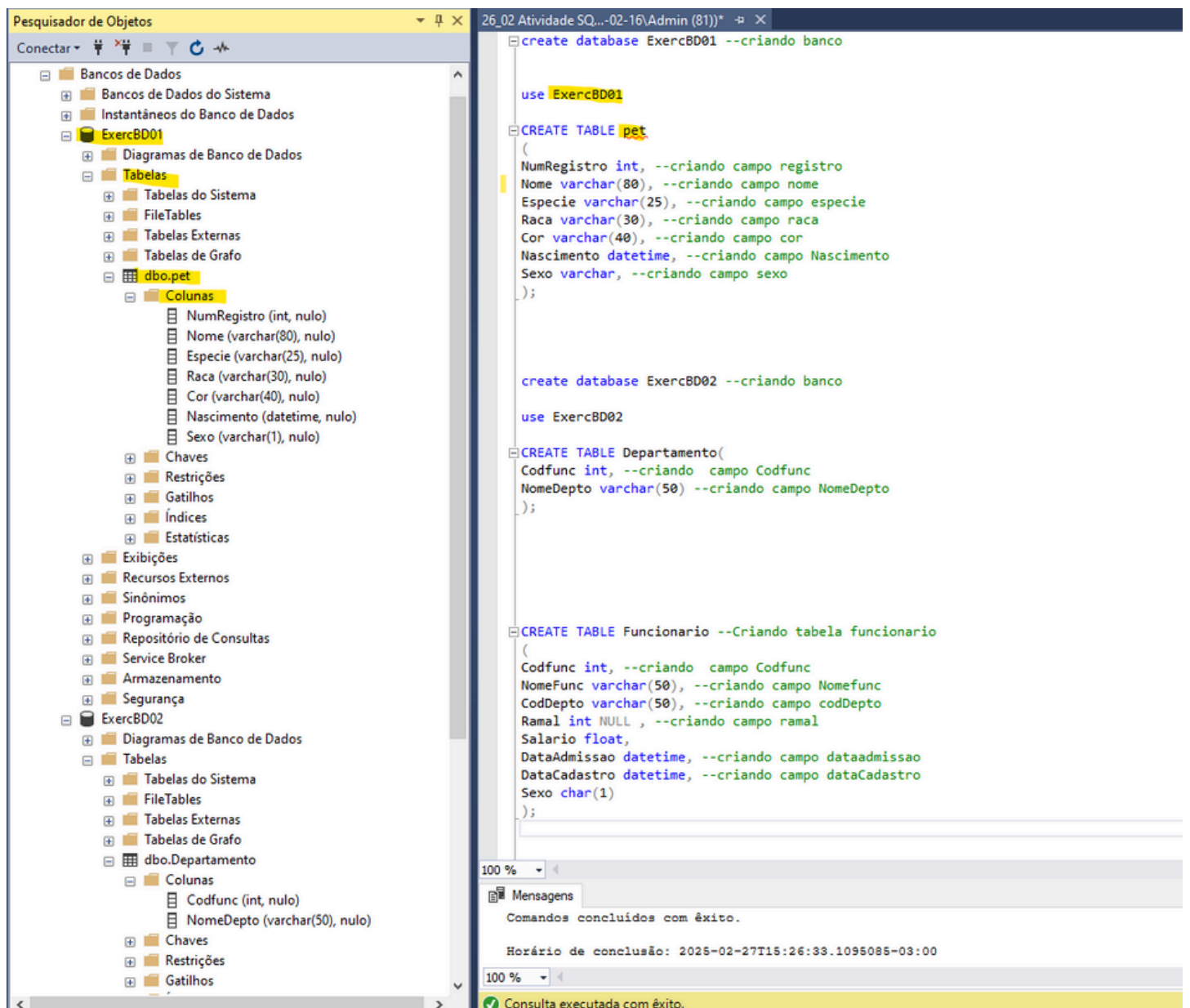
```
Raca varchar(30), --criando campo raca
```

```
Cor varchar(40), --criando campo cor
```

```
Nascimento datetime, --criando campo Nascimento
```

```
Sexo varchar, --criando campo sexo
```

```
);
```



## 2 - Criar um banco chamado "ExercBD02"

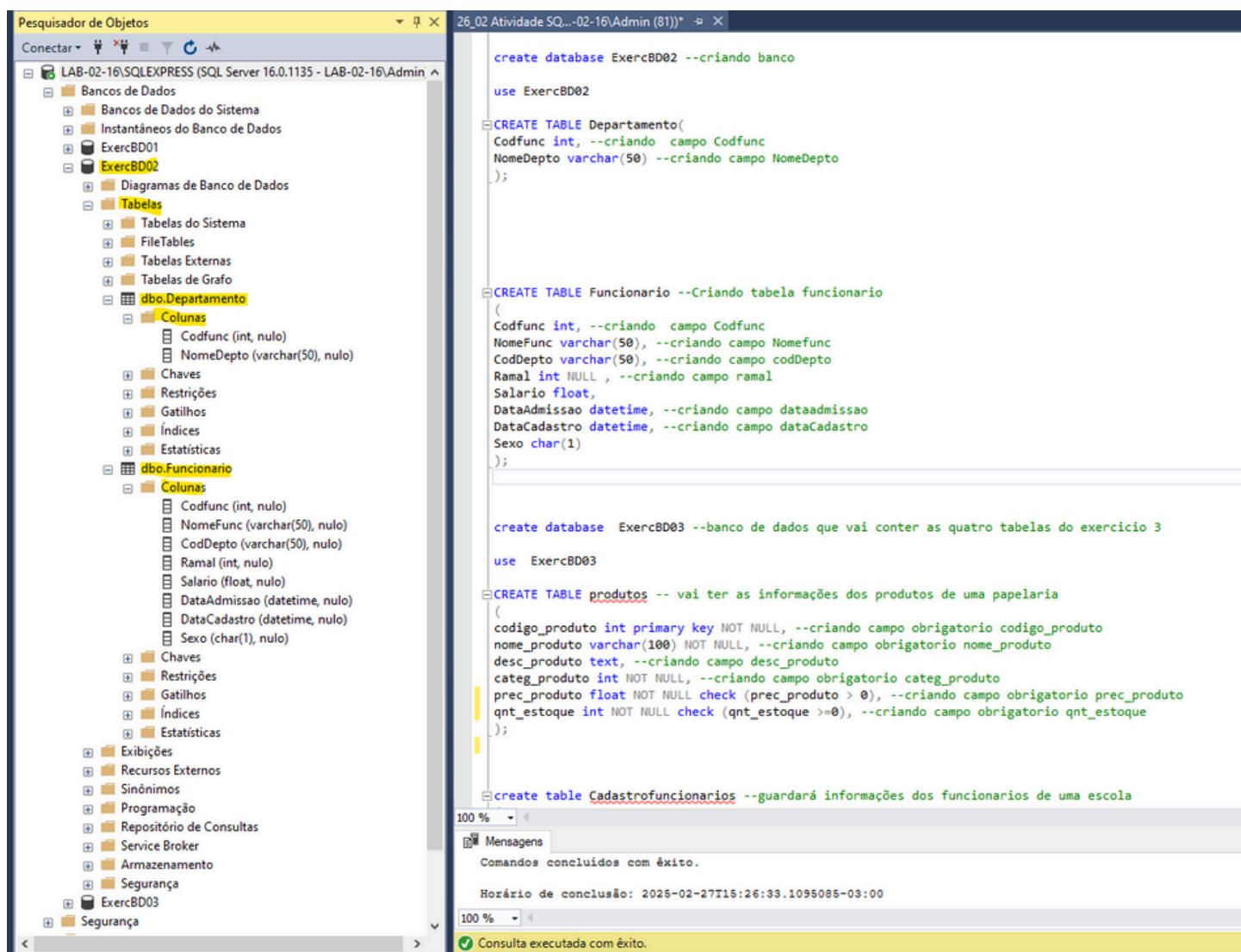
tabela 1 e tabela 2:

create database ExercBD02 --criando banco

use ExercBD02

```
CREATE TABLE Departamento(  
Codfunc int, --criando campo Codfunc  
NomeDepto varchar(50) --criando campo NomeDepto  
);
```

```
CREATE TABLE Funcionario --Criando tabela funcionario  
(  
Codfunc int, --criando campo Codfunc  
NomeFunc varchar(50), --criando campo Nomefunc  
CodDepto varchar(50), --criando campo codDepto  
Ramal int NULL , --criando campo ramal  
Salario float,  
DataAdmissao datetime, --criando campo dataadmissao  
DataCadastro datetime, --criando campo dataCadastro  
Sexo char(1)  
);
```



3.

## tabela 1 : produtos

create database ExercBD03 --banco de dados que vai conter as quatro tabelas do exercicio 3

use ExercBD03

CREATE TABLE produtos -- vai ter as informações dos produtos de uma papelaria

```
(  
codigo_produto int primary key NOT NULL, --criando campo  
obrigatorio codigo_produto  
  
nome_produto varchar(100) NOT NULL, --criando campo obrigatorio  
nome_produto  
  
desc_produto text, --criando campo desc_produto  
  
categ_produto int NOT NULL, --criando campo obrigatorio  
categ_produto  
  
prec_produto float NOT NULL check (prec_produto > 0), --criando  
campo obrigatorio prec_produto  
  
qnt_estoque int NOT NULL check (qnt_estoque >=0), --criando  
campo obrigatorio qnt_estoque  
  
);
```

## **tabela 2 : funcionario**

create table Cadastrofuncionarios --guardará informações dos  
funcionarios de uma escola

```
(  
  
IDFuncionario int primary key identity, --Identificador único do  
funcionário  
  
NomeFuncionario varchar (100) NOT NULL, --Nome completo do  
funcionário
```

CPF varchar(11) UNIQUE NOT NULL, --Cadastro de Pessoa Física (único)

Cargo varchar(50) NOT NULL, --Cargo ocupado pelo funcionário

Departamento\_ID int NOT NULL, --identificador que referencia o departamento

Salario float NOT NULL, --Valor do salário do funcionário

Data\_admissao date NOT NULL, --Data de contratação do funcionário

Telefone varchar(15) NOT NULL, -- Número de telefone para contato

Email Varchar (100) NOT NULL --Endereço de e-mail do funcionário

);

### **tabela 3 : fornecedores**

create table Fornecedores--Armazenará as informações dos Fornecedores de uma empresa de tecido

(

ID\_Fornecedor int primary key identity,--Identificador único do fornecedor

Nome\_Empresa varchar(100) NOT NULL,--Nome da empresa fornecedora

CNPJ varchar(14) unique NOT NULL, --Cadastro Nacional da Pessoa Jurídica

Endereco varchar(255), --Endereço da empresa fornecedora

Telefone varchar(15) NOT NULL, --Número de contato da empresa

Email varchar(100), --E-mail de contato

Categoria varchar(50) NOT NULL, --Tipo de material fornecido

);

The screenshot displays the SQL Server Enterprise Manager interface on the left and the SQL Server Enterprise Editor on the right. The Enterprise Manager shows the database structure for 'ExercBD03', including tables like 'Funcionarios', 'Departamentos', 'Produtos', and 'Fornecedores'. The Enterprise Editor shows the SQL code used to create these tables. The code includes comments in Portuguese explaining the purpose of each table and the data types used for each column. The 'Funcionarios' table has columns for ID, name, CPF, cargo, department, salary, admission date, phone, and email. The 'Departamentos' table has columns for ID, name, and address. The 'Produtos' table has columns for code, name, description, category, price, and stock. The 'Fornecedores' table has columns for ID, name, CNPJ, address, phone, email, and category. The code is executed successfully, as indicated by the 'Comandos concluídos com êxito.' message at the bottom.

```
create database ExercBD03 --banco de dados que vai conter as quatro tabelas do exercicio 3
use ExercBD03

CREATE TABLE produtos -- vai ter as informações dos produtos de uma papelaria
(
    codigo_produto int primary key NOT NULL, --criando campo obrigatorio codigo_produto
    nome_produto varchar(100) NOT NULL, --criando campo obrigatorio nome_produto
    desc_produto text, --criando campo desc_produto
    categ_produto int NOT NULL, --criando campo obrigatorio categ_produto
    prec_produto float NOT NULL check (prec_produto > 0), --criando campo obrigatorio prec_produto
    qnt_estoque int NOT NULL check (qnt_estoque >=0), --criando campo obrigatorio qnt_estoque
);

create table Cadastrofuncionarios --guardará informações dos funcionarios de uma escola
(
    IDFuncionario int primary key identity, --Identificador único do funcionário
    NomeFuncionario varchar (100) NOT NULL, --Nome completo do funcionário
    CPF varchar(11) UNIQUE NOT NULL, --Cadastro de Pessoa Física (único)
    Cargo varchar(50) NOT NULL, --Cargo ocupado pelo funcionário
    Departamento_ID int NOT NULL, --identificador que referencia o departamento
    Salario float NOT NULL, --Valor do salário do funcionário
    Data_admissao date NOT NULL, --Data de contratação do funcionário
    Telefone varchar(15) NOT NULL, -- Número de telefone para contato
    Email Varchar (100) NOT NULL --Endereço de e-mail do funcionário
);

create table Fornecedores--Armazenará as informações dos Fornecedores de uma empresa de tecido
(
    ID_Fornecedor int primary key identity,--Identificador único do fornecedor
    Nome_Empresa varchar(100) NOT NULL,--Nome da empresa fornecedora
    CNPJ varchar(14) unique NOT NULL, --Cadastro Nacional da Pessoa Jurídica
    Endereco varchar(255), --Endereço da empresa fornecedora
    Telefone varchar(15) NOT NULL, --Número de contato da empresa
    Email varchar(100), --E-mail de contato
    Categoria varchar(50) NOT NULL, --Tipo de material fornecido
);
```

Tabela Departamentos do exercício 3 e uma tabela extra:

tabela Departamentos:

```
create table Departamentos --Cadastro de departamentos de uma empresa
(
ID_Departamento int primary key identity, --Identificador único do departamento
Nome_Departamento varchar(100) NOT NULL, --Nome do departamento
Responsavel_ID int NOT NULL, --Identificador único do Responsavel pelo Departamento
Descricao varchar(100), --Breve descrição sobre o departamento
);
```

### **tabela mercado:**

```
create table mercado --tabela que tera as informações do mercado
(
nome_mercado varchar(20) NOT NULL, --nome do mercado
CNPJ varchar(14) unique NOT NULL, --CNPJ do mercado
endereco varchar(255), -- Endereço do mercado
Telefone varchar (15), --Telefone do mercado
produtos varchar(500) --produtos do mercado
);
```



Pesquisador de Objetos

Conectar ▾

LAB-02-16\SQLEXPRESS (SQL Server 16.0.1135 - LAB-02-16\Admin)

- Bancos de Dados
  - Bancos de Dados do Sistema
  - Instantâneos do Banco de Dados
  - ExercBD01
  - ExercBD02
  - ExercBD03
    - Diagramas de Banco de Dados
    - Tabelas
      - Tabelas do Sistema
      - FileTables
      - Tabelas Externas
      - Tabelas de Grafo
      - dbo.CadastroFuncionarios
      - dbo.Departamentos
        - Colunas
          - ID\_Departamento (PK, int, não nulo)
          - Nome\_Departamento (varchar(100), não nulo)
          - Responsavel\_ID (int, não nulo)
          - Descricao (varchar(100), nulo)
        - Chaves
        - Restrições
        - Gatilhos
        - Índices
        - Estatísticas
      - dbo.Fornecedores
      - dbo.mercado
        - Colunas
          - nome\_mercado (varchar(20), não nulo)
          - CNPJ (varchar(14), não nulo)
          - endereco (varchar(255), nulo)
          - Telefone (varchar(15), nulo)
          - produtos (varchar(500), nulo)
        - Chaves
        - Restrições
        - Gatilhos
        - Índices
        - Estatísticas
      - dbo.produtos
    - Exibições
    - Recursos Externos
    - Sinônimos
    - Programação
    - Repositório de Consultas
    - Service Broker
    - Armazenamento
    - Segurança

26\_02 Atividade SQ...-02-16\Admin (81))

```
create table Departamentos --Cadastro de departamentos de uma empresa
(
    ID_Departamento int primary key identity, --Identificador único do departamento
    Nome_Departamento varchar(100) NOT NULL, --Nome do departamento
    Responsavel_ID int NOT NULL, --Identificador único do Resposavel pelo Departamento
    Descricao varchar(100), --Breve descrição sobre o departamento
);

create table mercado --tabela que tera as informações do mercado
(
    nome_mercado varchar(20) NOT NULL, --nome do mercado
    CNPJ varchar(14) unique NOT NULL, --CNPJ do mercado
    endereco varchar(255), -- Endereço do mercado
    Telefone varchar (15), --Telefone do mercado
    produtos varchar(500) --produtos do mercado
);
```

100 % ▾

Mensagens

Comandos concluídos com êxito.

100 % ▾

✓ Consulta executada com êxito.

Pronto

Li 110