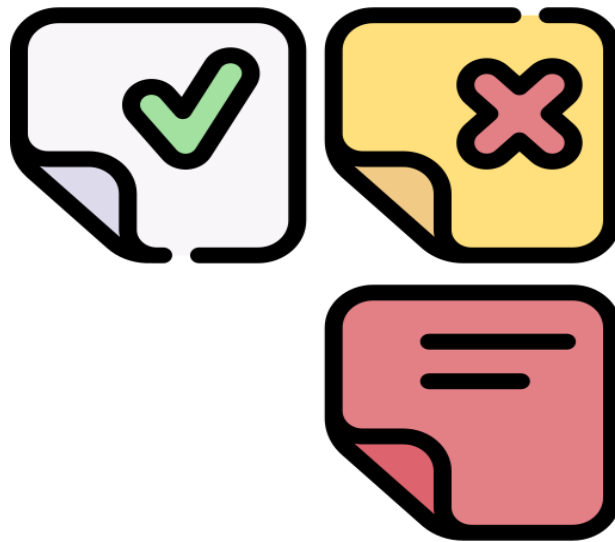


Aluno: GABRIELLY NASCIMENTO BENTO
Professor (a): EDNA RODRIGUES FERNANDES

ATIVIDADE DE BANCO DE DADOS II

TÍTULO



BANCO DE DADOS COM AS RESPECTIVAS TABELAS:

```
CREATE DATABASE BD_Comercio;
USE BD_Comercio;

CREATE TABLE Estado_Civil(
    Cod_estado_civ Int PRIMARY KEY,
    Desc_estado_civ Varchar(100) NOT NULL,
);
CREATE TABLE Cliente(
    Cod_Cliente Int PRIMARY KEY,
    Nome_Cliente Varchar(100) NOT NULL,
    Cod_estado_civ int NOT NULL,
    Salario Float NOT NULL,

    FOREIGN KEY(Cod_estado_civ) REFERENCES Estado_Civil,
);
CREATE TABLE Tipo_Fone(
    Cod_Fone Int PRIMARY KEY,
    Desc_Fone Varchar(100) NOT NULL,
);
CREATE TABLE Telefone(
    Cod_Cliente Int NOT NULL,
    Cod_Fone Int NOT NULL,
    Numero_Fone Varchar(12) NOT NULL,

    FOREIGN KEY(Cod_Fone) REFERENCES Tipo_Fone,
    FOREIGN KEY(Cod_Cliente) REFERENCES Cliente,
);
CREATE TABLE Func(
    Cod_Func Int PRIMARY KEY,
    Nome_Func Varchar(100) NOT NULL,
);
CREATE TABLE Pedido(
    Cod_Pedido Int PRIMARY KEY,
    Cod_Cliente Int NOT NULL,
    Cod_Func Int NOT NULL,
    Data_pedido DATE NOT NULL,

    FOREIGN KEY(Cod_Cliente) REFERENCES Cliente,
    FOREIGN KEY(Cod_Func) REFERENCES Func,
);
CREATE TABLE Produto(
    Cod_Produto Int PRIMARY KEY,
    Nome_Produto Varchar(100) NOT NULL,
    Tipo_Produto Varchar(100) NOT NULL,
);
CREATE TABLE Item_Pedido(
    Cod_Pedido Int NOT NULL,
    Cod_Produto Int NOT NULL,
    Qtde_Produto Int NOT NULL,

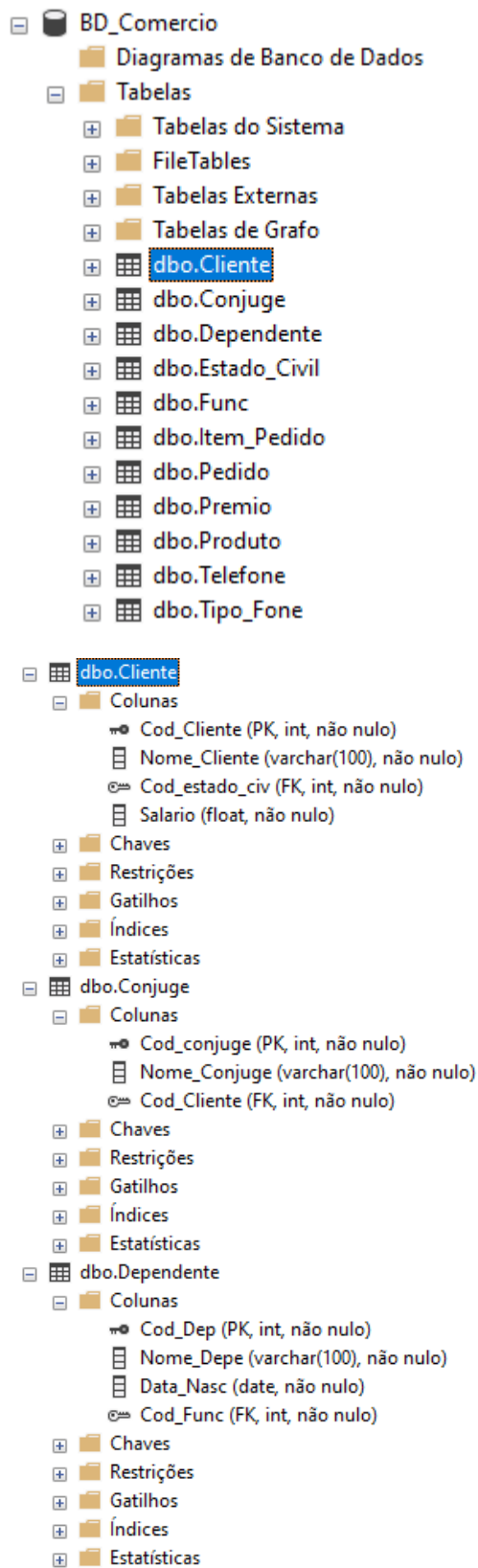
    FOREIGN KEY(Cod_Pedido) REFERENCES Pedido,
    FOREIGN KEY(Cod_Produto) REFERENCES Produto,
);
CREATE TABLE Premio(
    Cod_Func Int NOT NULL,
    Valor_Premio Float NOT NULL,

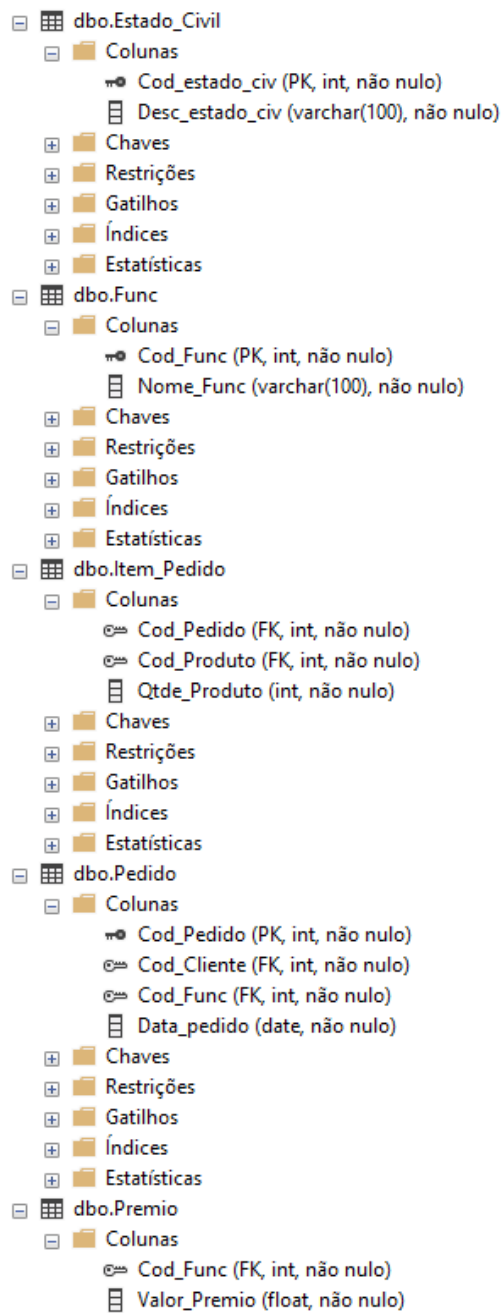
    FOREIGN KEY(Cod_Func) REFERENCES Func,
);
CREATE TABLE Dependente(
    Cod_Dep Int PRIMARY KEY,
    Nome_Depe Varchar(100) NOT NULL,
    Data_Nasc DATE NOT NULL,
    Cod_Func Int NOT NULL,






























    FOREIGN KEY(Cod_Func) REFERENCES Func,
);
CREATE TABLE Conjuge(
    Cod_conjuge Int PRIMARY KEY,
    Nome_Conjuge Varchar(100) NOT NULL,
    Cod_Cliente Int NOT NULL,

    FOREIGN KEY(Cod_Cliente) REFERENCES Cliente,
);
```

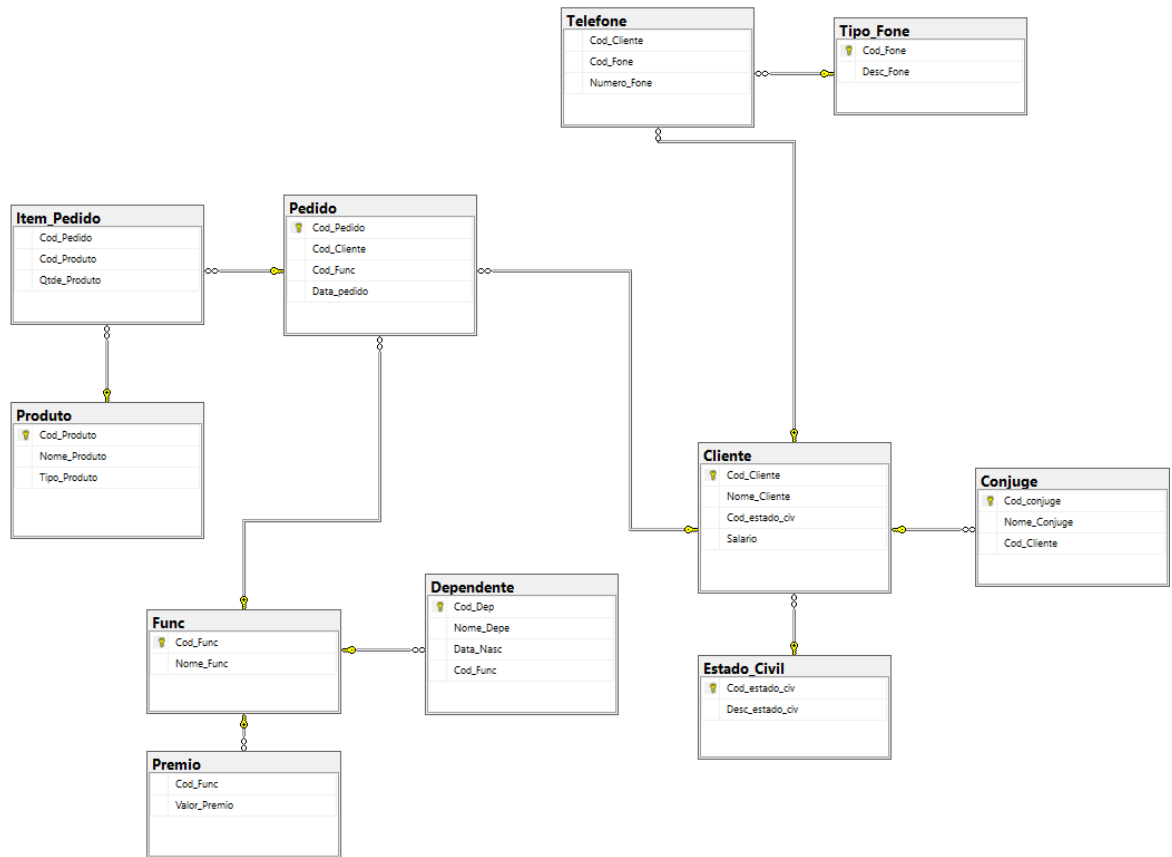
TABELAS:





- [-]  **dbo.Produto**
 - [-]  Colunas
 -  Cod_Produto (PK, int, não nulo)
 -  Nome_Produto (varchar(100), não nulo)
 -  Tipo_Produto (varchar(100), não nulo)
 - [+]  Chaves
 - [+]  Restrições
 - [+]  Gatilhos
 - [+]  Índices
 - [+]  Estatísticas
- [-]  **dbo.Telefone**
 - [-]  Colunas
 -  Cod_Cliente (FK, int, não nulo)
 -  Cod_Fone (FK, int, não nulo)
 -  Numero_Fone (varchar(12), nulo)
 - [+]  Chaves
 - [+]  Restrições
 - [+]  Gatilhos
 - [+]  Índices
 - [+]  Estatísticas
- [-]  **dbo.Tipo_Fone**
 - [-]  Colunas
 -  Cod_Fone (PK, int, não nulo)
 -  Desc_Fone (varchar(20), não nulo)
 - [+]  Chaves
 - [+]  Restrições
 - [+]  Gatilhos
 - [+]  Índices
 - [+]  Estatísticas

RELACIONAMENTO:



INSERÇÃO DE DADOS EM TODAS AS TABELAS (DE ACORDO COM OS ENUNCIADOS):

```
-- INSERT --
INSERT INTO Cliente Values(1, 'Daniel Lima', 1, '1200.00');
INSERT INTO Cliente Values(2, 'Rener Lopez', 2, '1200.00');
INSERT INTO Cliente Values(3, 'João Bieber', 1, '2000.00');
INSERT INTO Cliente Values(4, 'Maria Silvia', 3, '2000.00');
INSERT INTO Cliente Values(5, 'Karinne Ferreira', 2, '1200.00');
INSERT INTO Cliente Values(6, 'Bernardo Farias', 1, '2000.00');
SELECT * FROM Cliente;

INSERT INTO Telefone Values(1, 2, '99999-1111');
INSERT INTO Telefone Values(2, 1, '88888-2222');
INSERT INTO Telefone Values(3, 2, '99999-3333');
INSERT INTO Telefone Values(4, 3, '77777-4444');
INSERT INTO Telefone Values(5, 2, '66666-5555');
INSERT INTO Telefone Values(6, 4, 'NULL');
SELECT * FROM Telefone;

INSERT INTO Tipo_Fone Values (1, 'Residencial');
INSERT INTO Tipo_Fone Values (2, 'Celular');
INSERT INTO Tipo_Fone Values (3, 'Comercial');
INSERT INTO Tipo_Fone Values (4, 'Não Possui Telefone');
SELECT * FROM Tipo_Fone;

INSERT INTO Estado_Civil Values(1, 'Casado');
INSERT INTO Estado_Civil Values(2, 'Solteiro');
INSERT INTO Estado_Civil Values(3, 'Separado');
SELECT * FROM Estado_Civil;

INSERT INTO Func Values(1, 'Francisco');
INSERT INTO Func Values(2, 'Roseane');
INSERT INTO Func Values(3, 'Marcos');
INSERT INTO Func Values(4, 'Leticia');
INSERT INTO Func Values(5, 'Jovana');
SELECT * FROM Func;

INSERT INTO Produto Values(1, 'Fósforo', 'Doméstico');
INSERT INTO Produto Values(2, 'Vela', 'Doméstico');
INSERT INTO Produto Values(3, 'Feijão', 'Alimento');
INSERT INTO Produto Values(4, 'Arroz', 'Alimento');
INSERT INTO Produto Values(5, 'Leite', 'Alimento');
INSERT INTO Produto Values(6, 'Café', 'Alimento');
INSERT INTO Produto Values(7, 'Açúcar', 'Alimento');
INSERT INTO Produto Values(8, 'Sal', 'Alimento');
INSERT INTO Produto VALUES (9, 'Macarrão', 'Alimento');
SELECT * FROM Produto;

INSERT INTO Pedido Values(1, 1, 1, '2025-01-10');
INSERT INTO Pedido Values(2, 2, 2, '2025-01-12');
INSERT INTO Pedido Values(3, 3, 3, '2025-01-05');
INSERT INTO Pedido Values(4, 4, 1, '2025-02-01');
INSERT INTO Pedido Values(5, 5, 2, '2025-02-10');
INSERT INTO Pedido Values(6, 6, 4, '2025-02-15');
SELECT * FROM Pedido;

INSERT INTO Item_Pedido Values(1, 1, 2);
INSERT INTO Item_Pedido Values(1, 2, 2);
INSERT INTO Item_Pedido Values(2, 3, 3);
INSERT INTO Item_Pedido Values(2, 1, 1);
INSERT INTO Item_Pedido Values(3, 4, 2);
INSERT INTO Item_Pedido Values(3, 5, 1);
INSERT INTO Item_Pedido Values(4, 6, 1);
INSERT INTO Item_Pedido Values(5, 7, 1);
INSERT INTO Item_Pedido Values(6, 8, 2);
SELECT * FROM Item_Pedido;

INSERT INTO Premio Values(1, '500.00');
INSERT INTO Premio Values(2, '200.00');
INSERT INTO Premio Values(3, '300.00');
INSERT INTO Premio Values(4, '100.00');
SELECT * FROM Premio;

INSERT INTO Dependente Values(1, 'Lucas Kleber', '2010-05-05', 1);
INSERT INTO Dependente Values(2, 'Maria Kleber', '2012-03-15', 1);
INSERT INTO Dependente Values(3, 'Felipe Park', '2018-07-20', 2);
INSERT INTO Dependente Values(4, 'Helena Curuça', '2020-10-12', 3);
SELECT * FROM Dependente;

INSERT INTO Conjuge Values(1, 'Clara Lima', 1); -- Daniel
INSERT INTO Conjuge Values(2, 'Julia Bieber', 3); -- Joao
INSERT INTO Conjuge Values(3, 'Beatriz Farias', 6); -- Bernardo
SELECT * FROM Conjuge;
```

UTILIZAR OS COMANDOS SELECTS RESPECTIVOS PARA EFETUAR AS CONSULTAS ABAIXO:

Parte 1:

1 - Selecione o nome dos clientes e o número de todos os telefones que cada cliente possui:

```
SELECT Cliente.Nome_Cliente, Telefone.Numero_Fone
FROM Cliente
INNER JOIN Telefone
ON Cliente.Cod_Cliente = Telefone.Cod_Cliente;
```

Nome_Cliente	Numero_Fone
Daniel Lima	99999-1111
Renner Lopez	88888-2222
João Bieber	99999-3333
Maria Silvia	77777-4444
Karinne Ferreira	66666-5555
Bernardo Farias	NULL

2 - Selecione o nome dos clientes casados e o nome de seus cônjuges:

```
SELECT Cliente.Nome_Cliente, Conjuge.Nome_Conjuge
FROM Cliente
INNER JOIN Conjuge
ON Cliente.Cod_Cliente = Conjuge.Cod_Cliente;
```

Nome_Cliente	Nome_Conjuge
Daniel Lima	Clara Lima
João Bieber	Julia Bieber
Bernardo Farias	Beatriz Farias

3 - Selecione o nome dos clientes, o número e o tipo de telefone que cada um possui:

```
SELECT Cliente.Nome_Cliente, Telefone.Numero_Fone, Tipo_Fone.Desc_Fone
FROM Cliente
INNER JOIN Telefone ON Cliente.Cod_Cliente = Telefone.Cod_Cliente
INNER JOIN Tipo_Fone ON Telefone.Cod_Fone = Tipo_Fone.Cod_Fone;
```

Nome_Cliente	Numero_Fone	Desc_Fone
Daniel Lima	99999-1111	Celular
Renner Lopez	88888-2222	Residencial
João Bieber	99999-3333	Celular
Maria Silvia	77777-4444	Comercial
Karinne Ferreira	66666-5555	Celular
Bernardo Farias	NULL	Não Possui Telefone

4 - Selecione todas as colunas da tabela pedido, o nome do cliente que fez o pedido e o nome do funcionário que atendeu cada pedido:

```
SELECT Pedido.*, Cliente.Nome_Cliente, Func.Nome_Func
FROM Pedido
INNER JOIN Cliente ON Pedido.Cod_Cliente = Cliente.Cod_Cliente
INNER JOIN Func ON Pedido.Cod_Func = Func.Cod_Func;
```

Cod_Pedido	Cod_Cliente	Cod_Func	Data_pedido	Nome_Cliente	Nome_Func
1	1	1	2025-01-10	Daniel Lima	Francisco Kleber
2	2	2	2025-01-12	Rener Lopez	Roseane Park
3	3	3	2025-01-05	João Bieber	Marcos Curuça
4	4	1	2025-02-01	Maria Silvia	Francisco Kleber
5	5	2	2025-02-10	Karinne Ferreira	Roseane Park
6	6	4	2025-02-15	Bernardo Farias	Leticia Oliveira

5 - Selecione o código e a data do pedido, o nome do cliente que fez o pedido do funcionário “Francisco”:

```
SELECT Pedido.Cod_Pedido, Pedido.Data_pedido, Cliente.Nome_Cliente
FROM Pedido
INNER JOIN Cliente ON Pedido.Cod_Cliente = Cliente.Cod_Cliente
INNER JOIN Func ON Pedido.Cod_Func = Func.Cod_Func
WHERE Func.Nome_Func = 'Francisco Kleber';
```

Cod_Pedido	Data_pedido	Nome_Cliente
1	2025-01-10	Daniel Lima
4	2025-02-01	Maria Silvia

6 - Selecione o código e a data do pedido, o nome do funcionário que atendeu o pedido do cliente “Rener”:

```
SELECT Pedido.Cod_Pedido, Pedido.Data_pedido, Func.Nome_Func
FROM Pedido
INNER JOIN Cliente ON Pedido.Cod_Cliente = Cliente.Cod_Cliente
INNER JOIN Func ON Pedido.Cod_Func = Func.Cod_Func
WHERE Cliente.Nome_Cliente = 'Rener Lopez';
```

Cod_Pedido	Data_pedido	Nome_Func
2	2025-01-12	Roseane Park

Parte 2:

7 - Mostre o nome e a data de nascimento dos dependentes de cada funcionário:

```
SELECT Dependente.Nome_Depe, Dependente.Data_Nasc, Func.Nome_Func
FROM Dependente
INNER JOIN Func
ON Dependente.Cod_Func = Func.Cod_Func;
```

Nome_Depe	Data_Nasc	Nome_Func
Lucas Kleber	2010-05-05	Francisco Kleber
Maria Kleber	2012-03-15	Francisco Kleber
Felipe Park	2018-07-20	Roseane Park
Helena Curuça	2020-10-12	Marcos Curuça

8 - Selecione o código e a data do pedido e o nome de cada produto vendido:

```
SELECT Pedido.Cod_Pedido, Pedido.Data_pedido, Produto.Nome_Produto
FROM Pedido
INNER JOIN Item_Pedido ON Pedido.Cod_Pedido = Item_Pedido.Cod_Pedido
INNER JOIN Produto ON Produto.Cod_Produto = Item_Pedido.Cod_Produto ;
```

Cod_Pedido	Data_pedido	Nome_Produto
1	2025-01-10	Fósforo
1	2025-01-10	Vela
2	2025-01-12	Feijão
2	2025-01-12	Fósforo
3	2025-01-05	Aroz
3	2025-01-05	Leite
4	2025-02-01	Café
5	2025-02-10	Açucar
6	2025-02-15	Sal

9 - Selecione o código e a data do pedido e o nome de funcionário que vendeu “Fósforo”:

```
SELECT Pedido.Cod_Pedido, Pedido.Data_pedido, Func.Nome_Func
FROM Pedido
INNER JOIN Func ON Pedido.Cod_Func = Func.Cod_Func
INNER JOIN Item_Pedido ON Pedido.Cod_Pedido = Item_Pedido.Cod_Pedido
INNER JOIN Produto ON Produto.Cod_Produto = Item_Pedido.Cod_Produto ;
WHERE Produto.Nome_Produto = 'Fósforo';
```

Cod_Pedido	Data_pedido	Nome_Func
1	2025-01-10	Francisco Kleber
2	2025-01-12	Roseane Park

10 - Selecione o código e a data do pedido e o nome dos produtos comprados pelo cliente “Daniel”:

```
SELECT Pedido.Cod_Pedido, Pedido.Data_pedido, Produto.Nome_Produto
FROM Pedido
INNER JOIN Cliente ON Pedido.Cod_Cliente = Cliente.Cod_Cliente
INNER JOIN Item_Pedido ON Pedido.Cod_Pedido = Item_Pedido.Cod_Pedido
INNER JOIN Produto ON Produto.Cod_Produto = Item_Pedido.Cod_Produto
WHERE Cliente.Nome_Cliente = 'Daniel Lima';
```

Cod_Pedido	Data_pedido	Nome_Produto
1	2025-01-10	Fósforo
1	2025-01-10	Vela

11 - Selecione todos os produtos vendidos pela funcionária “Roseane”:

```
SELECT Produto.Nome_Produto
FROM Produto
INNER JOIN Item_Pedido ON Produto.Cod_Produto = Item_Pedido.Cod_Produto
INNER JOIN Pedido ON Item_Pedido.Cod_Pedido = Pedido.Cod_Pedido
INNER JOIN Func ON Pedido.Cod_Func = Func.Cod_Func
WHERE Func.Nome_Func = 'Roseane Park';
```

Nome_Produto
Feijão
Fósforo
Açúcar

12 - Selecione o nome dos clientes e o nome dos produtos comprados respectivamente:

```
SELECT Cliente.Nome_Cliente, Produto.Nome_Produto
FROM Cliente
INNER JOIN Pedido ON Cliente.Cod_Cliente = Pedido.Cod_Cliente
INNER JOIN Item_Pedido ON Pedido.Cod_Pedido = Item_Pedido.Cod_Pedido
INNER JOIN Produto ON Produto.Cod_Produto = Item_Pedido.Cod_Produto;
```

Nome_Cliente	Nome_Produto
Daniel Lima	Fósforo
Daniel Lima	Vela
Rener Lopez	Feijão
Rener Lopez	Fósforo
João Bieber	Arroz
João Bieber	Leite
Maria Silvia	Café
Karinne Ferr...	Açúcar
Bernardo Fa...	Sal

13 - Selecione o nome dos funcionários e o nome dos produtos vendidos respectivamente:

```

SELECT Func.Nome_Func, Produto.Nome_Produto
FROM Func
INNER JOIN Pedido ON Func.Cod_Func = Pedido.Cod_Func
INNER JOIN Item_Pedido ON Pedido.Cod_Pedido = Item_Pedido.Cod_Pedido
INNER JOIN Produto ON Produto.Cod_Produto = Item_Pedido.Cod_Produto;

```

Nome_Func	Nome_Produto
Francisco Kleber	Fósforo
Francisco Kleber	Vela
Roseane Park	Feijão
Roseane Park	Fósforo
Marcos Curuça	Aroz
Marcos Curuça	Leite
Francisco Kleber	Café
Roseane Park	Açucar
Leticia Oliveira	Sal

Parte 3:

14 - Mostre o nome dos funcionários e o valor total dos prêmios que cada funcionário tem:

```

SELECT Func.Nome_Func, SUM(premio.Valor_Premio) AS valor_total
FROM Func
INNER JOIN Premio ON Func.Cod_Func = Premio.Cod_Func
GROUP BY Func.Nome_Func;

```

Nome_Func	valor_total
Francisco Kleber	500
Leticia Oliveira	100
Marcos Curuça	300
Roseane Park	200

15 - Mostre o nome dos funcionários e quantidade de dependentes de cada funcionário:

```

SELECT Func.Nome_Func, COUNT(Dependente.Cod_Dep) AS total_dep
FROM Func
INNER JOIN Dependente ON Func.Cod_Func = Dependente.Cod_Func
GROUP BY Func.Nome_Func;

```

Nome_Func	total_dep
Francisco Kleber	2
Marcos Curuça	1
Roseane Park	1

16 - Mostre a quantidade de clientes “Casados”, “Solteiros” e “Separados”:

```

SELECT Cod_estado_civ, COUNT(*) AS Total
FROM Cliente
GROUP BY Cod_estado_civ;

```

Cod_estado_civ	Total
1	3
2	2
3	1

17 - Selecione os dados dos clientes que não tem telefone:

```
SELECT *
FROM Cliente
INNER JOIN Telefone ON Cliente.Cod_Cliente = Telefone.Cod_Cliente
WHERE Telefone.Cod_Fone = 4;
```

Cod_Cliente	Nome_Cliente	Cod_estado_civ	Salario	Cod_Cliente	Cod_Fone	Numero_Fone
6	Bernardo Farias	1	2000	6	4	NULL

18 - Selecione os dados dos clientes “Solteiros”:

```
SELECT Cliente.Cod_Cliente, Cliente.Nome_Cliente, Cliente.Cod_estado_civ, Cliente.Salario
FROM Cliente
INNER JOIN Estado_Civil ON Cliente.Cod_estado_civ = Estado_Civil.Cod_estado_civ
WHERE Estado_Civil.Cod_estado_civ = 2;
```

Cod_Cliente	Nome_Cliente	Cod_estado_civ	Salario
2	Rener Lopez	2	1200
5	Karinne Ferreira	2	1200

19 - Selecione os dados dos clientes “Casados”:

```
SELECT Cliente.Cod_Cliente, Cliente.Nome_Cliente, Cliente.Cod_estado_civ, Cliente.Salario
FROM Cliente
INNER JOIN Estado_Civil ON Cliente.Cod_estado_civ = Estado_Civil.Cod_estado_civ
WHERE Estado_Civil.Cod_estado_civ = 1;
```

Cod_Cliente	Nome_Cliente	Cod_estado_civ	Salario
1	Daniel Lima	1	1200
3	João Bieber	1	2000
6	Bernardo Farias	1	2000

20 - Selecione os dados dos funcionários que não têm prêmios:

```
SELECT *
FROM Func
WHERE Cod_Func
NOT IN (SELECT Cod_Func FROM Premio WHERE Cod_Func IS NOT NULL);
```

Cod_Func	Nome_Func
5	Jovana Kenel

21 - Selecione os dados dos funcionários que não têm dependentes:

```

SELECT *
FROM Func
WHERE Cod_Func
NOT IN (SELECT Cod_Func FROM Dependente WHERE Cod_Func IS NOT NULL);

```

Cod_Func	Nome_Func
4	Leticia Oliveira
5	Jovana Kenel

22 - Selecione os produtos que nunca foram vendidos:

```

SELECT *
FROM Produto
WHERE Cod_Produto
NOT IN (SELECT Cod_Produto FROM Item_Pedido WHERE Cod_Produto IS NOT NULL);

```

Cod_Produto	Nome_Produto	Tipo_Produto
9	Macarrão	Alimento