**PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ**

PROJETO DE BANCO DE DADOS

1ª PARTE

Panificadora Padoka

Equipe:

Vinicius Costa

Vinicius Viana

Marcos VInicius

Rafael Galafassi

Curitiba

Setembro de 2023

Sumário

[1. Domínio de Aplicação para o Banco de Dados 3](#_Toc143529170)

[1.1. Nome do Projeto 3](#_Toc143529171)

[1.2. Tema do Projeto 3](#_Toc143529172)

[1.3. Papéis / Usuários que o sistema identifica 3](#_Toc143529173)

[1.4. 1ª Funcionalidade do Projeto 3](#_Toc143529174)

[2. Modelo Conceitual 3](#_Toc143529176)

[3. Modelo Lógico (normalizado até a 2FN) 3](#_Toc143529177)

[4. Modelo Físico (normalizado até a 2FN) 3](#_Toc143529178)

[4.1. SQL para criação de tabelas e restrições (DDL) 3](#_Toc143529179)

[4.2. SQL para inserção de pelo menos 10 registros para cada tabela (DML) 3](#_Toc143529180)

[4.3. SQL para 2 consultas simples de recuperação de dados (DML) 3](#_Toc143529181)

[4.4. SQL para 2 consultas de recuperação de dados, envolvendo PK e FK (DML) 3](#_Toc143529182)

# Domínio de Aplicação para o Banco de Dados

## Panificadora Padoca

## Tema do projeto

A área de negócio escolhida para o projeto é uma panificadora, um estabelecimento da area de comercio e alimentício, que produz e vende uma variedade de produtos de panificação, como pães, bolos, doces, salgados e outros itens relacionados.

O projeto tem como objetivo conter um sistema de venda, estoque, e gerenciamento de panificação.

## Papéis / Usuários que o sistema identifica

**Ambiente externo do Cliente:**

Clientes

**Ambiente interno do Lojista:**

Fornecedores

## 1ª Registro de funcionalidades do projeeto

* Realização de pedidos a Padaria
* Atendimento dos pedidos pelos Funcionários
* Controle de pedidos
* Controle de estoque de produtos

# Modelo Conceitual

# 

**Entidades:**

Clientes

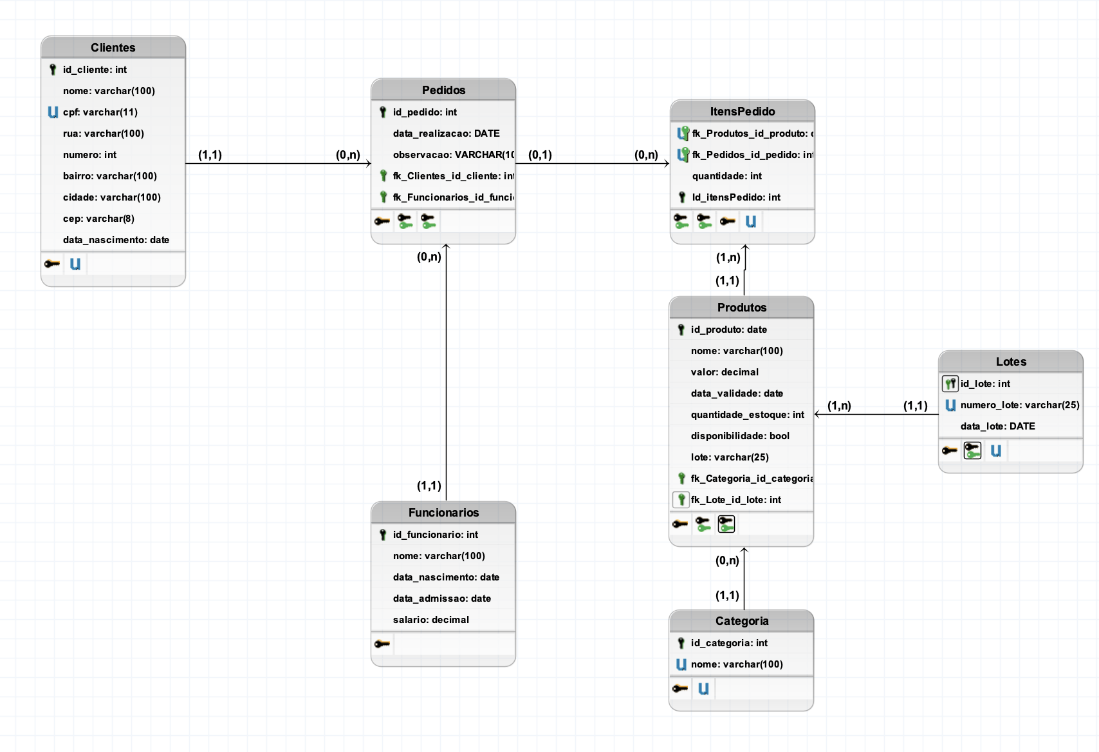
Pedidos

Produtos

Categorias

Funcionários

# Modelo Lógico (normalizado até a 2FN)



OBS: Gerou uma nova entidade do relacionamento muitos para muitos de pedidos com produtos.

# Modelo Físico (normalizado até a 2FN)

## SQL para criação de tabelas e restrições (DDL)

CREATE TABLE Clientes (

id\_cliente int PRIMARY KEY,

nome varchar(100),

cpf varchar(11) UNIQUE,

rua varchar(100),

numero int,

bairro varchar(100),

cidade varchar(100),

cep varchar(8),

data\_nascimento date

);

CREATE TABLE Pedidos (

id\_pedido int PRIMARY KEY,

data\_realizacao DATE,

observacao VARCHAR(1000),

fk\_Clientes\_id\_cliente int,

fk\_Funcionarios\_id\_funcionario int

);

CREATE TABLE Produtos (

id\_produto date PRIMARY KEY,

nome varchar(100),

valor decimal,

data\_validade date,

quantidade\_estoque int,

disponibilidade bool,

lote varchar(25),

fk\_Categoria\_id\_categoria int,

fk\_Lote\_id\_lote int

);

CREATE TABLE Funcionarios (

id\_funcionario int PRIMARY KEY,

nome varchar(100),

data\_nascimento date,

data\_admissao date,

salario decimal

);

CREATE TABLE Categoria (

id\_categoria int PRIMARY KEY,

nome varchar(100) UNIQUE

);

CREATE TABLE ItensPedido (

fk\_Produtos\_id\_produto date,

fk\_Pedidos\_id\_pedido int,

quantidade int,

Id\_itensPedido int PRIMARY KEY,

UNIQUE (fk\_Produtos\_id\_produto, fk\_Pedidos\_id\_pedido)

);

CREATE TABLE Lotes (

id\_lote int PRIMARY KEY,

numero\_lote varchar(25) UNIQUE,

data\_lote DATE

);

ALTER TABLE Pedidos ADD CONSTRAINT FK\_Pedidos\_2

FOREIGN KEY (fk\_Clientes\_id\_cliente)

REFERENCES Clientes (id\_cliente)

ON DELETE CASCADE;

ALTER TABLE Pedidos ADD CONSTRAINT FK\_Pedidos\_3

FOREIGN KEY (fk\_Funcionarios\_id\_funcionario)

REFERENCES Funcionarios (id\_funcionario)

ON DELETE CASCADE;

ALTER TABLE Produtos ADD CONSTRAINT FK\_Produtos\_2

FOREIGN KEY (fk\_Categoria\_id\_categoria)

REFERENCES Categoria (id\_categoria)

ON DELETE CASCADE;

ALTER TABLE Produtos ADD CONSTRAINT FK\_Produtos\_3

FOREIGN KEY (fk\_Lote\_id\_lote)

REFERENCES Lotes (id\_lote);

ALTER TABLE ItensPedido ADD CONSTRAINT FK\_ItensPedido\_1

FOREIGN KEY (fk\_Produtos\_id\_produto)

REFERENCES Produtos (id\_produto)

ON DELETE RESTRICT;

ALTER TABLE ItensPedido ADD CONSTRAINT FK\_ItensPedido\_2

FOREIGN KEY (fk\_Pedidos\_id\_pedido)

REFERENCES Pedidos (id\_pedido)

ON DELETE SET NULL;

## SQL para inserção de pelo menos 10 registros para cada tabela (DML)

INSERT INTO Clientes (id\_cliente, nome, cpf, rua, numero, bairro, cidade, cep, data\_nascimento)

VALUES (1, 'Enrico Ricardo Bento da Conceição', '44733799225', 'Estrada Transamazônica, s/n', 214, 'Sucunduri', 'Apuí', '69275970', '2001-07-17');

INSERT INTO Clientes (id\_cliente, nome, cpf, rua, numero, bairro, cidade, cep, data\_nascimento)

VALUES (2, 'Lorenzo Mário Novaes', '63330296291', 'Estrada Transamazônica, s/n', 829, 'Sucunduri', 'Apuí', '69275970', '2001-09-10');

INSERT INTO Clientes (id\_cliente, nome, cpf, rua, numero, bairro, cidade, cep, data\_nascimento)

VALUES (3, 'Julio Cláudio Oliver Souza', '21587939290', 'Avenida 13 de Novembro 850', 872, 'Centro', 'Apuí', '69265970', '2001-01-13');

INSERT INTO Clientes (id\_cliente, nome, cpf, rua, numero, bairro, cidade, cep, data\_nascimento)

VALUES (4, 'Lorenzo Sérgio da Luz', '09927098227', 'Estrada Transamazônica, s/n', 853, 'Sucunduri', 'Apuí', '69275970', '2001-03-10');

INSERT INTO Clientes (id\_cliente, nome, cpf, rua, numero, bairro, cidade, cep, data\_nascimento)

VALUES (5, 'Francisco Diego Enzo Baptista', '70623119285', 'Estrada Transamazônica, s/n', 985, 'Sucunduri', 'Apuí', '69275970', '2001-08-25');

INSERT INTO Clientes (id\_cliente, nome, cpf, rua, numero, bairro, cidade, cep, data\_nascimento)

VALUES (6, 'Erick Fernando Pinto', '09057788209', 'Estrada Transamazônica, s/n', 668, 'Sucunduri', 'Apuí', '69275970', '2001-01-14');

INSERT INTO Clientes (id\_cliente, nome, cpf, rua, numero, bairro, cidade, cep, data\_nascimento)

VALUES (7, 'Thales Márcio Murilo Araújo', '90897248260', 'Avenida 13 de Novembro 850', 563, 'Centro', 'Apuí', '69265970', '2001-05-07');

INSERT INTO Clientes (id\_cliente, nome, cpf, rua, numero, bairro, cidade, cep, data\_nascimento)

VALUES (8, 'Matheus Augusto Porto', '07081493246', 'Estrada Transamazônica, s/n', 475, 'Sucunduri', 'Apuí', '69275970', '2001-01-04');

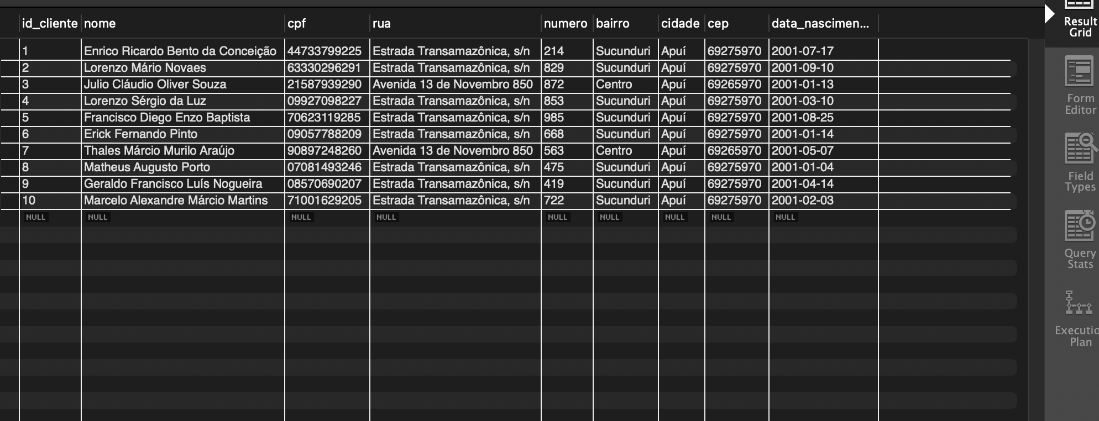
INSERT INTO Clientes (id\_cliente, nome, cpf, rua, numero, bairro, cidade, cep, data\_nascimento)

VALUES (9, 'Geraldo Francisco Luís Nogueira', '08570690207', 'Estrada Transamazônica, s/n', 419, 'Sucunduri', 'Apuí', '69275970', '2001-04-14');

INSERT INTO Clientes (id\_cliente, nome, cpf, rua, numero, bairro, cidade, cep, data\_nascimento)

VALUES (10, 'Marcelo Alexandre Márcio Martins', '71001629205', 'Estrada Transamazônica, s/n', 722, 'Sucunduri', 'Apuí', '69275970', '2001-02-03');

**Resultado:**



INSERT INTO Funcionarios (id\_funcionario, nome, data\_nascimento, data\_admissao, salario)

VALUES (1, 'Tomás Lorenzo Lopes', '2000-01-16', '2023-09-24', 3500.00);

INSERT INTO Funcionarios (id\_funcionario, nome, data\_nascimento, data\_admissao, salario)

VALUES (2, 'Sebastião Thales Assunção', '2000-04-05', '2023-09-24', 3200.00);

INSERT INTO Funcionarios (id\_funcionario, nome, data\_nascimento, data\_admissao, salario)

VALUES (3, 'Julio Rafael Martin Rodrigues', '2000-04-12', '2023-09-24', 3400.00);

INSERT INTO Funcionarios (id\_funcionario, nome, data\_nascimento, data\_admissao, salario)

VALUES (4, 'Filipe Márcio Davi Nogueira', '2000-04-26', '2023-09-24', 3600.00);

INSERT INTO Funcionarios (id\_funcionario, nome, data\_nascimento, data\_admissao, salario)

VALUES (5, 'Matheus Isaac Daniel Oliveira', '2000-04-11', '2023-09-24', 3300.00);

INSERT INTO Funcionarios (id\_funcionario, nome, data\_nascimento, data\_admissao, salario)

VALUES (6, 'Pedro Henrique Juan Castro', '2000-02-19', '2023-09-24', 3700.00);

INSERT INTO Funcionarios (id\_funcionario, nome, data\_nascimento, data\_admissao, salario)

VALUES (7, 'Bryan Diogo Araújo', '2000-02-08', '2023-09-24', 3100.00);

INSERT INTO Funcionarios (id\_funcionario, nome, data\_nascimento, data\_admissao, salario)

VALUES (8, 'Mateus Sebastião Theo Sales', '2000-06-13', '2023-09-24', 3800.00);

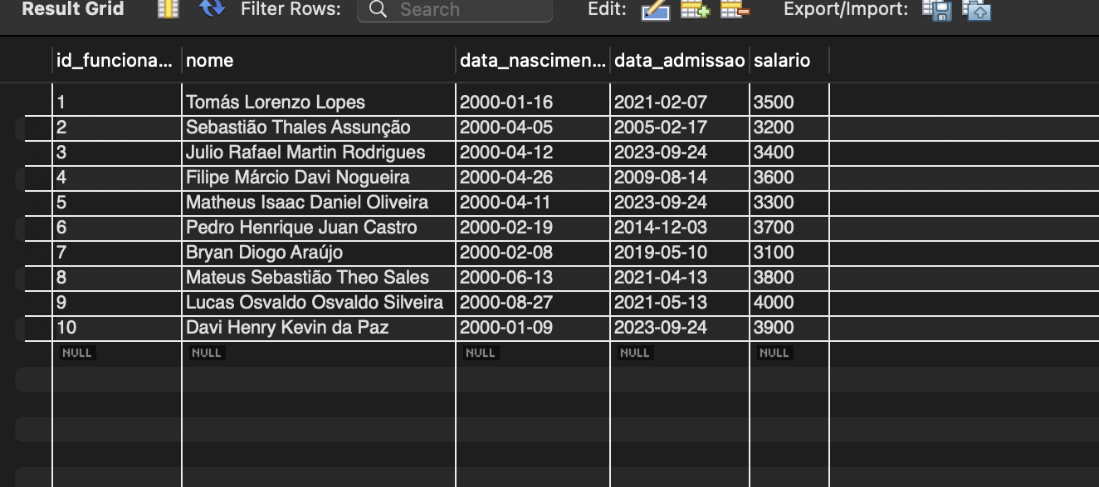
INSERT INTO Funcionarios (id\_funcionario, nome, data\_nascimento, data\_admissao, salario)

VALUES (9, 'Lucas Osvaldo Osvaldo Silveira', '2000-08-27', '2023-09-24', 4000.00);

INSERT INTO Funcionarios (id\_funcionario, nome, data\_nascimento, data\_admissao, salario)

VALUES (10, 'Davi Henry Kevin da Paz', '2000-01-09', '2023-09-24', 3900.00);

**Resultado:**



INSERT INTO Categoria (id\_categoria, nome)

VALUES (1, 'Pães');

INSERT INTO Categoria (id\_categoria, nome)

VALUES (2, 'Bolos');

INSERT INTO Categoria (id\_categoria, nome)

VALUES (3, 'Bolachas');

INSERT INTO Categoria (id\_categoria, nome)

VALUES (4, 'Doces');

INSERT INTO Categoria (id\_categoria, nome)

VALUES (5, 'Salgados');

**Resultado:**



INSERT INTO Lotes (id\_lote, numero\_lote, data\_lote)

VALUES (1, 'LOTE2023001', '2023-01-15');

INSERT INTO Lotes (id\_lote, numero\_lote, data\_lote)

VALUES (2, 'LOTE2023002', '2023-02-20');

INSERT INTO Lotes (id\_lote, numero\_lote, data\_lote)

VALUES (3, 'LOTE2023003', '2023-03-10');

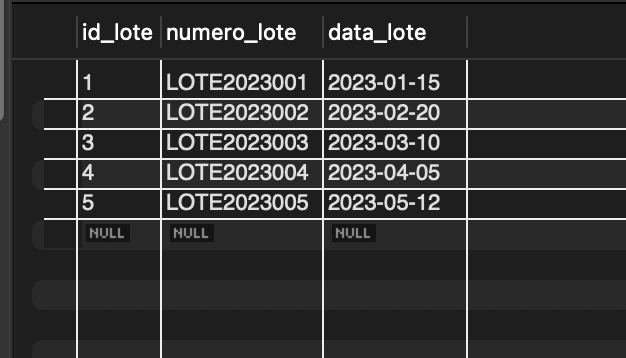
INSERT INTO Lotes (id\_lote, numero\_lote, data\_lote)

VALUES (4, 'LOTE2023004', '2023-04-05');

INSERT INTO Lotes (id\_lote, numero\_lote, data\_lote)

VALUES (5, 'LOTE2023005', '2023-05-12');

**Resultado:**



ALTER TABLE Produtos DROP COLUMN lote;

INSERT INTO Produtos (id\_produto, nome, valor, data\_validade, quantidade\_estoque, disponibilidade, fk\_Categoria\_id\_categoria, fk\_Lote\_id\_lote)

VALUES (1, 'Pão Francês', 1.50, '2023-10-10', 100, 1, 1, 1);

INSERT INTO Produtos (id\_produto, nome, valor, data\_validade, quantidade\_estoque, disponibilidade, fk\_Categoria\_id\_categoria, fk\_Lote\_id\_lote)

VALUES (2, 'Pão de Forma Integral', 2.00, '2023-10-12', 80, 1, 1, 2);

INSERT INTO Produtos (id\_produto, nome, valor, data\_validade, quantidade\_estoque, disponibilidade, fk\_Categoria\_id\_categoria, fk\_Lote\_id\_lote)

VALUES (3, 'Bolo de Cenoura', 8.00, '2023-10-15', 20, 1, 2, 3);

INSERT INTO Produtos (id\_produto, nome, valor, data\_validade, quantidade\_estoque, disponibilidade, fk\_Categoria\_id\_categoria, fk\_Lote\_id\_lote)

VALUES (4, 'Bolo de Chocolate', 8.00, '2023-10-15', 15, 1, 2, 3);

INSERT INTO Produtos (id\_produto, nome, valor, data\_validade, quantidade\_estoque, disponibilidade, fk\_Categoria\_id\_categoria, fk\_Lote\_id\_lote)

VALUES (5, 'Biscoito de Chocolate', 3.50, '2023-11-01', 50, 1, 3, 4);

INSERT INTO Produtos (id\_produto, nome, valor, data\_validade, quantidade\_estoque, disponibilidade, fk\_Categoria\_id\_categoria, fk\_Lote\_id\_lote)

VALUES (6, 'Biscoito de Coco', 3.50, '2023-11-01', 60, 1, 3, 4);

INSERT INTO Produtos (id\_produto, nome, valor, data\_validade, quantidade\_estoque, disponibilidade, fk\_Categoria\_id\_categoria, fk\_Lote\_id\_lote)

VALUES (7, 'Doce de Leite', 5.00, '2023-10-30', 30, 1, 4, 5);

INSERT INTO Produtos (id\_produto, nome, valor, data\_validade, quantidade\_estoque, disponibilidade, fk\_Categoria\_id\_categoria, fk\_Lote\_id\_lote)

VALUES (8, 'Brigadeiro', 2.50, '2023-10-28', 40, 1, 4, 5);

INSERT INTO Produtos (id\_produto, nome, valor, data\_validade, quantidade\_estoque, disponibilidade, fk\_Categoria\_id\_categoria, fk\_Lote\_id\_lote)

VALUES (9, 'Salgado de Queijo', 3.00, '2023-10-25', 25, 1, 5, 1);

INSERT INTO Produtos (id\_produto, nome, valor, data\_validade, quantidade\_estoque, disponibilidade, fk\_Categoria\_id\_categoria, fk\_Lote\_id\_lote)

VALUES (10, 'Salgado de Presunto', 3.00, '2023-10-25', 30, 1, 5, 2);

INSERT INTO Produtos (id\_produto, nome, valor, data\_validade, quantidade\_estoque, disponibilidade, fk\_Categoria\_id\_categoria, fk\_Lote\_id\_lote)

VALUES (11, 'Croissant', 2.00, '2023-10-20', 20, 1, 1, 3);

INSERT INTO Produtos (id\_produto, nome, valor, data\_validade, quantidade\_estoque, disponibilidade, fk\_Categoria\_id\_categoria, fk\_Lote\_id\_lote)

VALUES (12, 'Rosquinha', 1.50, '2023-10-25', 40, 1, 3, 4);

INSERT INTO Produtos (id\_produto, nome, valor, data\_validade, quantidade\_estoque, disponibilidade, fk\_Categoria\_id\_categoria, fk\_Lote\_id\_lote)

VALUES (13, 'Pão de Queijo', 2.50, '2023-10-15', 30, 1, 5, 3);

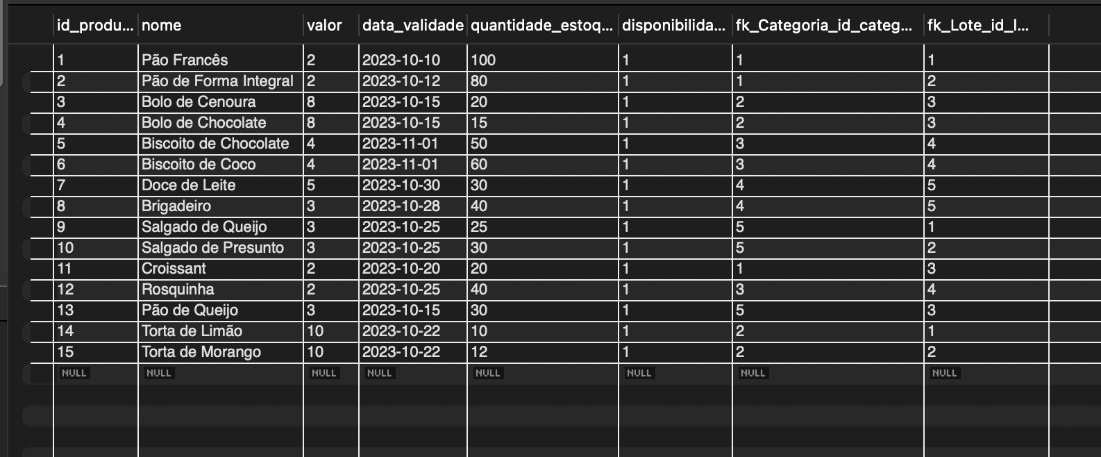
INSERT INTO Produtos (id\_produto, nome, valor, data\_validade, quantidade\_estoque, disponibilidade, fk\_Categoria\_id\_categoria, fk\_Lote\_id\_lote)

VALUES (14, 'Torta de Limão', 10.00, '2023-10-22', 10, 1, 2, 1);

INSERT INTO Produtos (id\_produto, nome, valor, data\_validade, quantidade\_estoque, disponibilidade, fk\_Categoria\_id\_categoria, fk\_Lote\_id\_lote)

VALUES (15, 'Torta de Morango', 10.00, '2023-10-22', 12, 1, 2, 2);

**Resultado:**



INSERT INTO Pedidos (id\_pedido, data\_realizacao, observacao, fk\_Clientes\_id\_cliente, fk\_Funcionarios\_id\_funcionario)

VALUES (1, '2023-09-24', 'Entregar na Rua A', 1, 1);

INSERT INTO ItensPedido (Id\_itensPedido, fk\_Produtos\_id\_produto, fk\_Pedidos\_id\_pedido, quantidade, valor\_unitario)

VALUES (1, 1, 1, 10, 1.50);

INSERT INTO ItensPedido (Id\_itensPedido, fk\_Produtos\_id\_produto, fk\_Pedidos\_id\_pedido, quantidade, valor\_unitario)

VALUES (2, 3, 1, 5, 8.00);

INSERT INTO Pedidos (id\_pedido, data\_realizacao, observacao, fk\_Clientes\_id\_cliente, fk\_Funcionarios\_id\_funcionario)

VALUES (2, '2023-09-25', 'Entregar na Rua B', 2, 2);

INSERT INTO ItensPedido (Id\_itensPedido, fk\_Produtos\_id\_produto, fk\_Pedidos\_id\_pedido, quantidade, valor\_unitario)

VALUES (3, 2, 2, 6, 2.00);

INSERT INTO ItensPedido (Id\_itensPedido, fk\_Produtos\_id\_produto, fk\_Pedidos\_id\_pedido, quantidade, valor\_unitario)

VALUES (4, 5, 2, 8, 3.50);

INSERT INTO Pedidos (id\_pedido, data\_realizacao, observacao, fk\_Clientes\_id\_cliente, fk\_Funcionarios\_id\_funcionario)

VALUES (3, '2023-09-26', 'Entregar na Rua C', 3, 3);

INSERT INTO ItensPedido (Id\_itensPedido, fk\_Produtos\_id\_produto, fk\_Pedidos\_id\_pedido, quantidade, valor\_unitario)

VALUES (5, 7, 3, 3, 5.00);

INSERT INTO ItensPedido (Id\_itensPedido, fk\_Produtos\_id\_produto, fk\_Pedidos\_id\_pedido, quantidade, valor\_unitario)

VALUES (6, 9, 3, 4, 3.00);

INSERT INTO Pedidos (id\_pedido, data\_realizacao, observacao, fk\_Clientes\_id\_cliente, fk\_Funcionarios\_id\_funcionario)

VALUES (4, '2023-09-27', 'Entregar na Rua D', 4, 4);

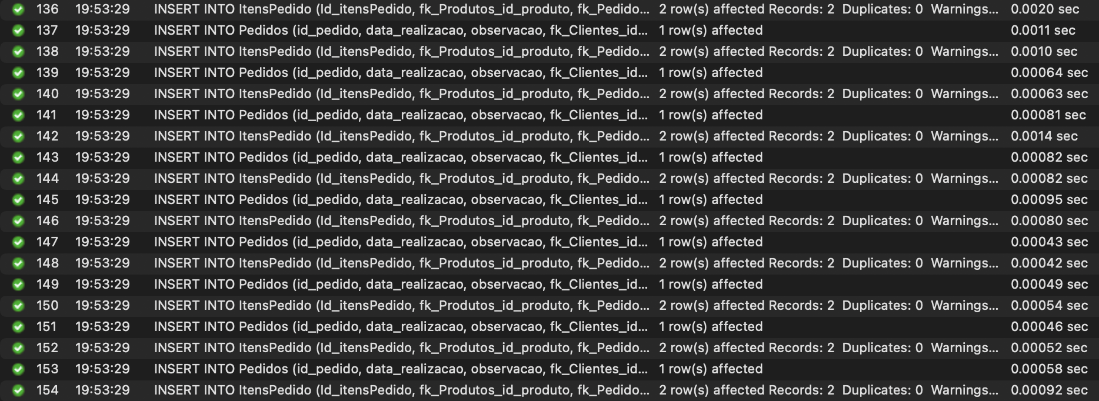
INSERT INTO ItensPedido (Id\_itensPedido, fk\_Produtos\_id\_produto, fk\_Pedidos\_id\_pedido, quantidade, valor\_unitario)

VALUES (7, 8, 4, 7, 2.50);

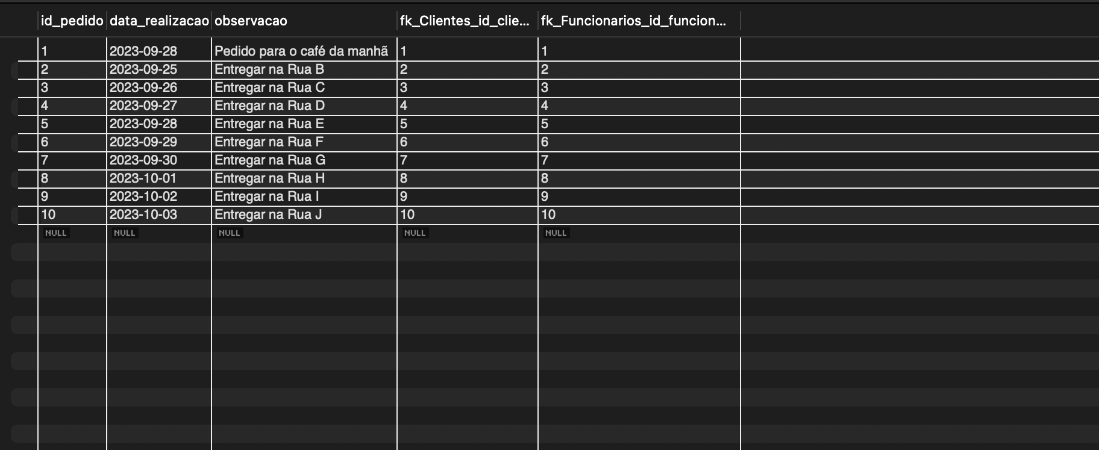
INSERT INTO ItensPedido (Id\_itensPedido, fk\_Produtos\_id\_produto, fk\_Pedidos\_id\_pedido, quantidade, valor\_unitario)

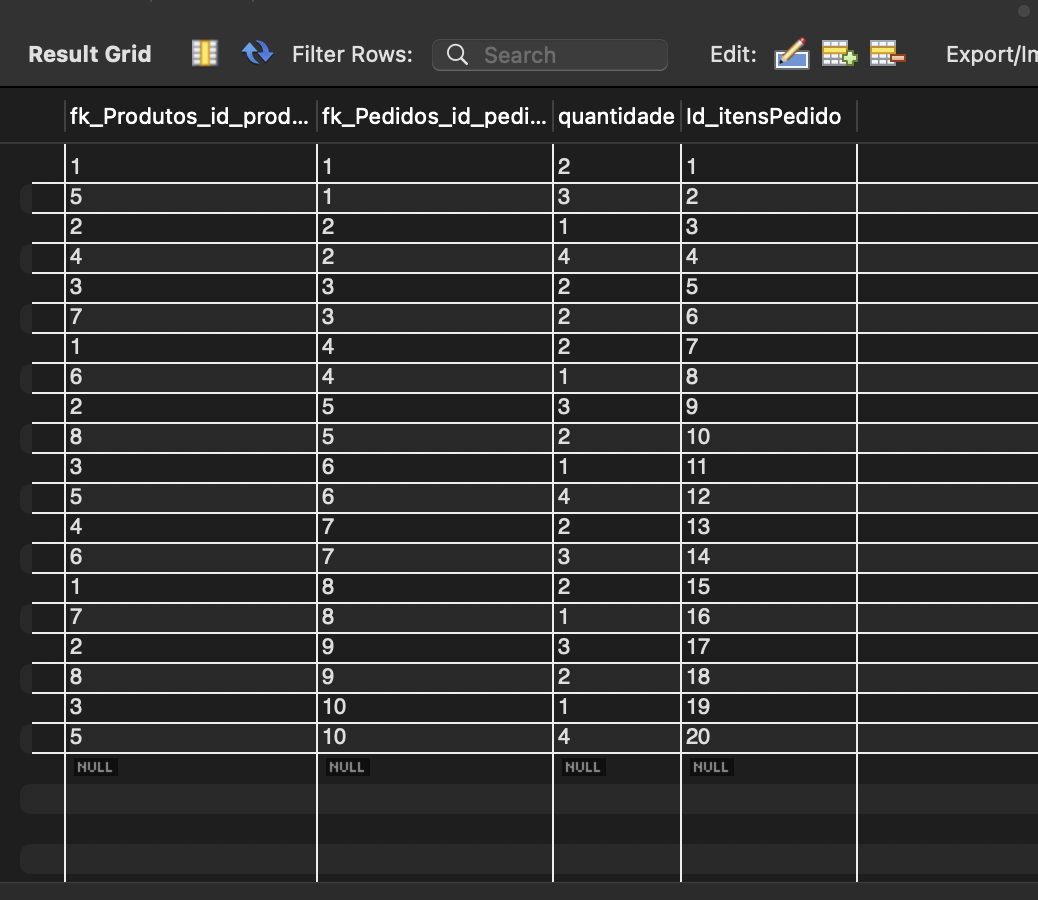
VALUES (8, 11, 4, 10, 2.00);

**Comprovação:**



**Resultado:**





## SQL para 2 consultas simples de recuperação de dados (DML)

1. Obter Todos os produtos em estoque, com a quantidade de dias para expirar a validade do produto.

SELECT

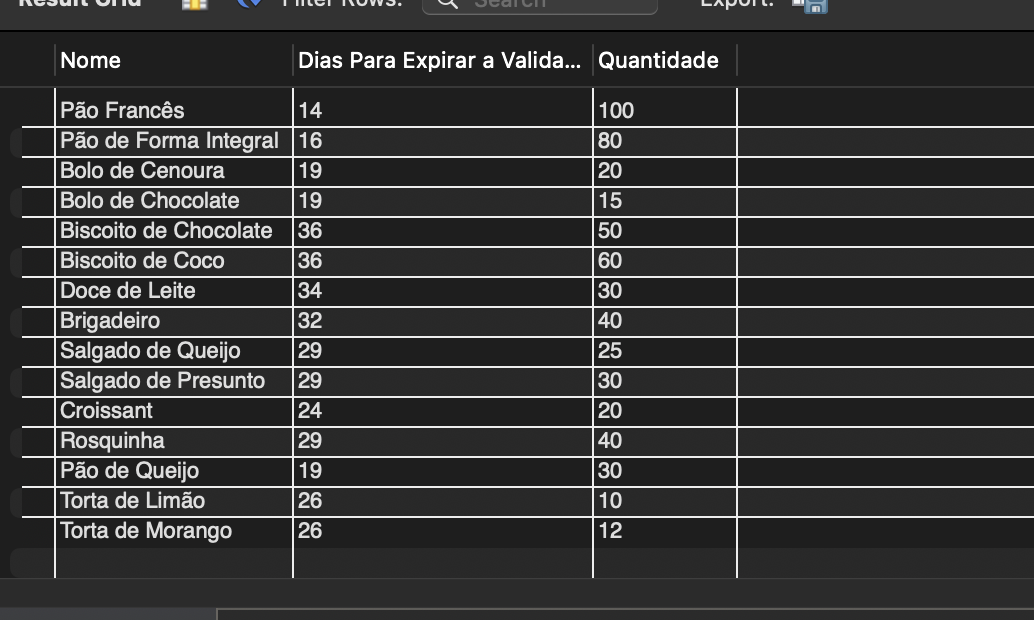
nome AS Nome,

DATEDIFF(data\_validade, CURDATE()) AS 'Dias Para Expirar a Validade',

quantidade\_estoque AS Quantidade

FROM Produtos

WHERE disponibilidade = 1;



* (extra) Média da data de validade por categoria:

SELECT

c.nome as Categoria,

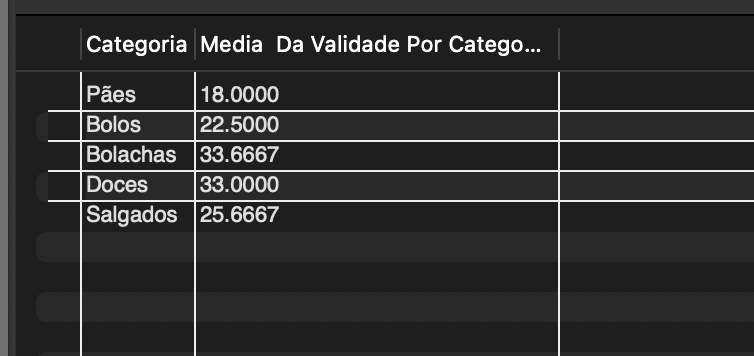
AVG(DATEDIFF(p.data\_validade, CURDATE())) AS 'Media Da Validade Por Categoria'

FROM Produtos p, Categorias c

WHERE p.fk\_Categoria\_id\_categoria = c.id\_categoria AND

disponibilidade = 1

GROUP BY p.fk\_Categoria\_id\_categoria;



1. Obter todos os funcionários ativos, seu tempo trabalhado e o valor que ele recebeu durante todo o tempo de trabalho (levando em conta o último salário informado).

SELECT

nome AS Nome,

TIMESTAMPDIFF(YEAR, data\_admissao, CURDATE()) AS AnosTrabalhados,

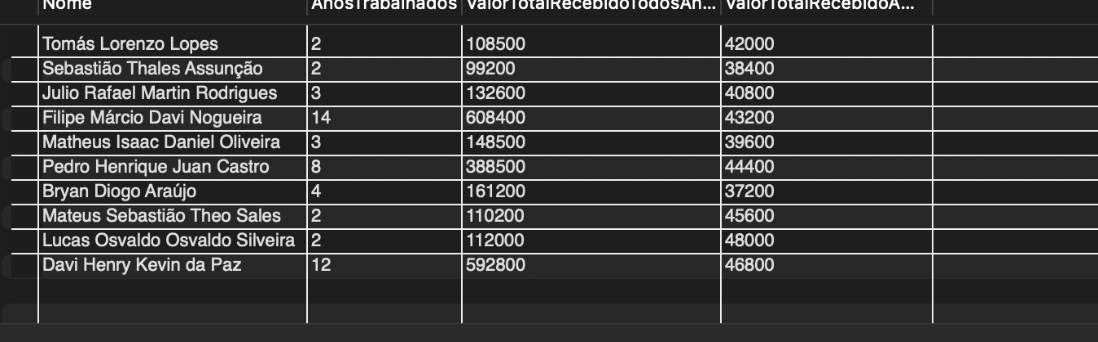
ROUND(

TIMESTAMPDIFF(MONTH, data\_admissao, CURDATE()) \* salario,

2) AS ValorTotalRecebidoTodosAnos,

ROUND(12 \* salario, 2) AS ValorTotalRecebidoAno

FROM Funcionarios;

;

## SQL para 2 consultas de recuperação de dados, envolvendo PK e FK (DML)

1. Obter todos os produtos que possui mais pedidos, categorias com mais pedidos, e lote com mais pedidos.

SELECT

C.nome AS Categoria,

SUM(I.quantidade) AS PedidosRealizados,

(SELECT P.nome

FROM Produtos AS P

JOIN ItensPedido AS IP ON P.id\_produto = IP.fk\_Produtos\_id\_produto

WHERE P.fk\_Categoria\_id\_categoria = C.id\_categoria

GROUP BY P.id\_produto

ORDER BY SUM(IP.quantidade) DESC

LIMIT 1) AS ProdutoMaisComprado,

(SELECT SUM(P.valor \* IP.quantidade)

FROM Produtos AS P

JOIN ItensPedido AS IP ON P.id\_produto = IP.fk\_Produtos\_id\_produto

WHERE P.fk\_Categoria\_id\_categoria = C.id\_categoria

) AS TotalEmVendas

FROM

Produtos AS P

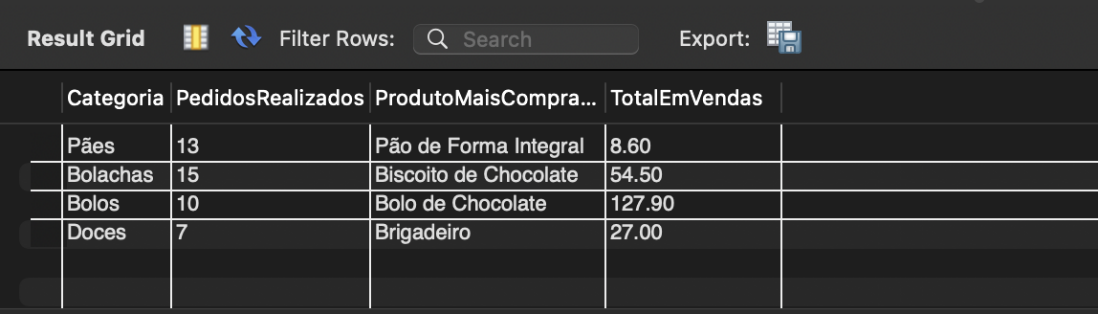
JOIN

ItensPedido AS I ON P.id\_produto = I.fk\_Produtos\_id\_produto

JOIN

Categorias AS C ON P.fk\_Categoria\_id\_categoria = C.id\_categoria

GROUP BY P.fk\_Categoria\_id\_categoria;



1. Obter Funcionários com mais pedidos atendidos, valor total vendido, e a média de vendas de cada funcionário.

SELECT

F.nome AS Funcionario,

COUNT(Pedido.TotalPedidos) AS TotalPedidosRealizados,

SUM(Pedido.valor\_total) AS ValorTotalVendas,

AVG(Pedido.valor\_total) AS MediaVendasPorPedido

FROM

Funcionarios AS F

JOIN

(

SELECT

P.fk\_Funcionarios\_id\_funcionario,

COUNT(P.id\_pedido) AS TotalPedidos,

SUM(IP.quantidade \* Pr.valor) AS valor\_total

FROM

Pedidos AS P

JOIN

ItensPedido AS IP ON P.id\_pedido = IP.fk\_Pedidos\_id\_pedido

JOIN

Produtos AS Pr ON IP.fk\_Produtos\_id\_produto = Pr.id\_produto

GROUP BY

P.fk\_Funcionarios\_id\_funcionario

) AS Pedido ON F.id\_funcionario = Pedido.fk\_Funcionarios\_id\_funcionario

GROUP BY

Funcionario

ORDER BY

TotalPedidosRealizados DESC;

