

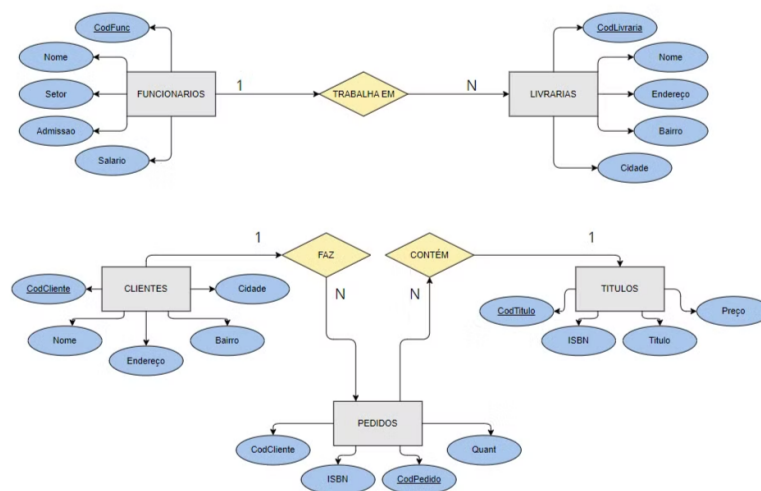
# Lista de Exercícios 02 - LE2

Claudio da Aparecida Meireles Filho - 2321070

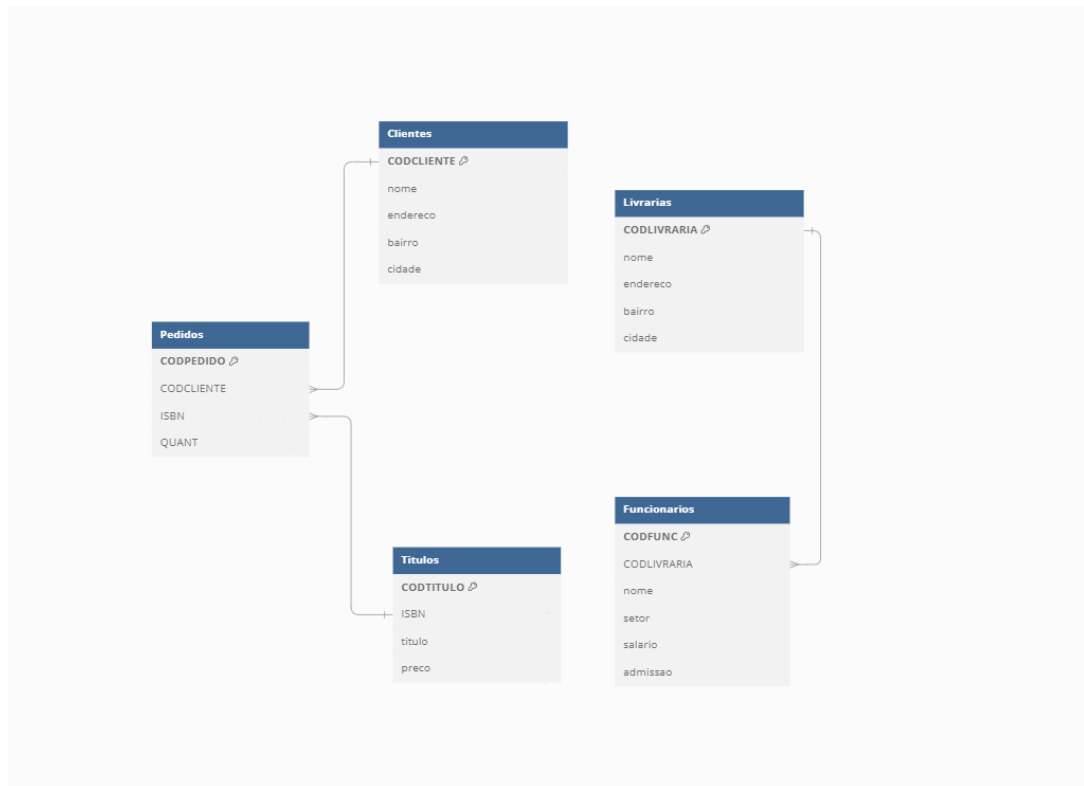
## Exercício 01:

Utilizando as tabelas abaixo, que representam dados para o caso de uso de uma livraria, elabore:

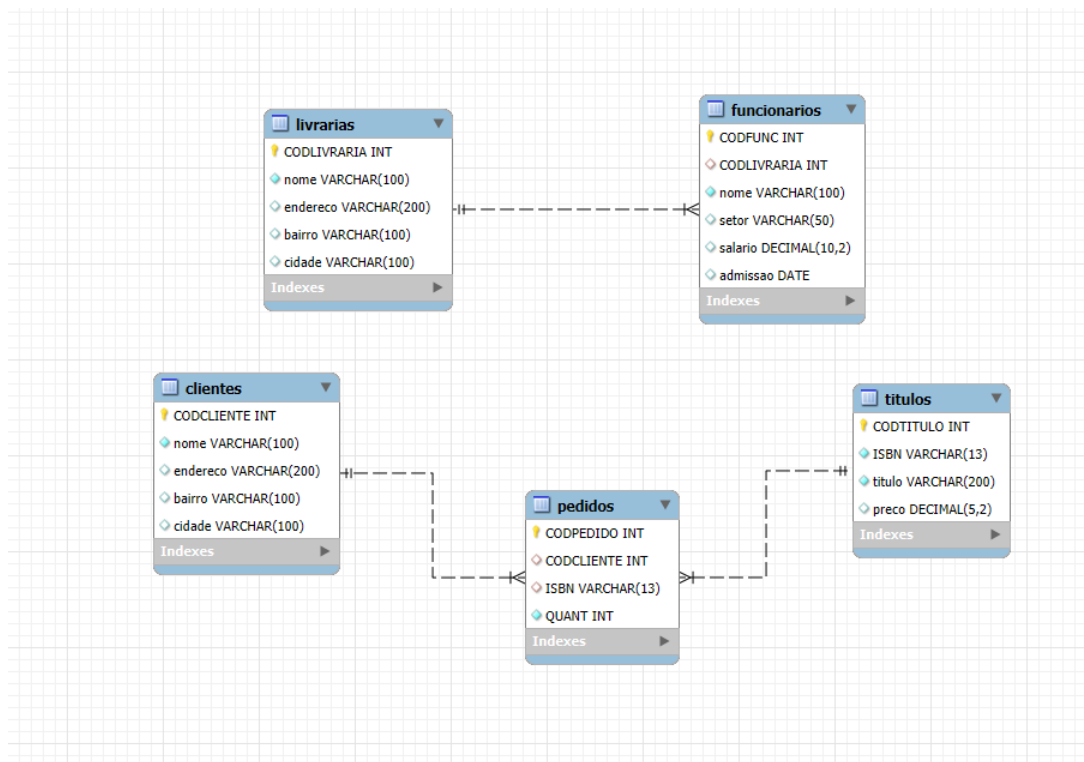
1. O modelo Conceitual no padrão MER (Modelo Entidade-Relacionamento) e realize o Diagrama Entidade-Relacionamento (DER) utilizando o modelo Chen;



2. O Modelo Lógico, utilizando o diagrama de Engenharia de Informação;



3. O Modelo Físico, especificando necessariamente os tipos de dados, tamanhos, domínios, chaves primárias, chaves estrangeiras e qualquer restrição necessária;



4. Scripts em SQL DDL para criação do banco de dados da Livraria e respectivas tabelas;

```

CREATE DATABASE livraria;

USE livraria;

-- Cria a tabela: Livrarias
CREATE TABLE Livrarias (
    CODLIVRARIA INT PRIMARY KEY, -- Chave primária
    nome VARCHAR(100) NOT NULL,
    endereco VARCHAR(200),
    bairro VARCHAR(100),
    cidade VARCHAR(100)
);

-- Cria a tabela: Clientes
CREATE TABLE Clientes (
    CODCLIENTE INT PRIMARY KEY, -- Chave primária
    nome VARCHAR(100) NOT NULL,
    endereco VARCHAR(200),
    bairro VARCHAR(100),
    cidade VARCHAR(100)
);

-- Cria a tabela: Titulos
CREATE TABLE Titulos (
    CODTITULO INT PRIMARY KEY AUTO_INCREMENT, -- Chave primária com auto incr
    ISBN VARCHAR(13) NOT NULL, -- Permite repetição do ISBN
    titulo VARCHAR(200) NOT NULL,
    preco DECIMAL(5, 2),
    INDEX idx_isbn (ISBN) -- Índice diretamente na coluna ISBN
);

-- Cria a tabela: Funcionarios
CREATE TABLE Funcionarios (
    CODFUNC INT PRIMARY KEY, -- Chave primária
    CODLIVRARIA INT, -- Chave estrangeira para Livrarias
    nome VARCHAR(100) NOT NULL,
    setor VARCHAR(50),
    salario DECIMAL(10, 2),
    admissao DATE,
    FOREIGN KEY (CODLIVRARIA) REFERENCES Livrarias(CODLIVRARIA) -- Definição
);

-- Cria a tabela: Pedidos
CREATE TABLE Pedidos (
    CODPEDIDO INT PRIMARY KEY AUTO_INCREMENT, -- Chave primária com auto incr
    CODCLIENTE INT, -- Chave estrangeira para Clientes
    ISBN VARCHAR(13), -- Chave estrangeira para Titulos

```

```

    QUANT INT NOT NULL,
    FOREIGN KEY (CODCLIENTE) REFERENCES Clientes(CODCLIENTE), -- Definição da
    FOREIGN KEY (ISBN) REFERENCES Titulos(ISBN) -- Definição da chave estrangeira
);

```

5. Scripts em SQL DML para inserção dos dados, conforme apresentados nas tabelas.

```

-- Inserindo Dados na Tabela Livrarias
INSERT INTO Livrarias (CODLIVRARIA, nome, endereco, bairro, cidade) VALUES
(0001, 'BookShelf', 'Al. Santiago, 280', 'Centro', 'Atibaia'),
(0002, 'Paper News', 'R. Santos Dumont, 88', 'Jd. Brasil', 'São Paulo'),
(0003, 'WPA Livros', 'Av. Paulista, 1050', 'Jd. Paulista', 'Atibaia'),
(0004, 'Livro de Cabeceira', 'R. da Saudade, 33', 'Bela Vista', 'Rio Claro');

-- Inserindo Dados na Tabela Funcionarios
INSERT INTO Funcionarios (CODFUNC, CODLIVRARIA, nome, setor, salario, admissao) VALUES
(001, 0001, 'Felipe Gomes', 'RH', 1800.00, '2010-08-01'),
(002, 0002, 'Norberto Silva', 'Financeiro', 2100.00, '1998-03-10'),
(003, 0003, 'Kátia Salles', 'Contabilidade', 1800.00, '2013-04-25'),
(004, 0001, 'Anselmo Vieira', 'Contabilidade', 1800.00, '2006-01-18'),
(005, 0002, 'Margarete Moreira', 'Vendas', 1300.00, '2011-05-22'),
(006, 0003, 'Reinaldo Gonçalves', 'Vendas', 1300.00, '2013-10-06'),
(007, 0004, 'Felipe Camargo', 'Vendas', 1800.00, '1998-08-18');

-- Inserindo Dados na Tabela Clientes
INSERT INTO Clientes (CODCLIENTE, nome, endereco, bairro, cidade) VALUES
(0001, 'Jorge Almeida', 'R. da Saudade, 33', 'Centro', 'Atibaia'),
(0002, 'Roberto Alves', 'R. XV de Novembro, 200', 'Pq. das Américas', 'Bragança'),
(0003, 'Juvencio Bueno', 'Av. Santos Dumont, 350', 'Centro', 'Joanópolis'),
(0004, 'Marcos Rodrigues', 'Av. São João, 230', 'Jd. Paulista', 'Atibaia'),
(0005, 'André Ramos', 'Rua Vitória, 151', 'Bela Vista', 'São Paulo'),
(0006, 'Rute Barbosa', 'Rua Constantino, 200', 'Tatuapé', 'São Paulo'),
(0007, 'Aline Barros', 'Al. Barão de Limeira, 105', 'Vila Alemã', 'Rio Claro');

-- Inserindo Dados na Tabela Titulos
INSERT INTO Titulos (ISBN, titulo, preco) VALUES
('9781234567890', 'Computação e Animação Gráfica com Blender 2.80', 89.00),
('978234567893', 'Kotlin na Prática', 85.00),
('9782234567891', 'Introdução à Programação de Computadores', 68.00),
('978234567893', 'Kotlin na Prática', 85.00),
('978234567892', 'Desenvolvimento para Web', 60.00),
('9782234567891', 'Introdução à Programação de Computadores', 68.00),
('978234567893', 'Kotlin na Prática', 85.00);

-- Inserindo Dados na Tabela Pedidos
INSERT INTO Pedidos (ISBN, QUANT, CODCLIENTE) VALUES
('9781234567890', 1, 0002),

```

```
( '978234567893', 1, 0002),
( '9782234567891', 1, 0004),
( '978234567893', 1, 0004),
( '978234567892', 1, 0003),
( '9782234567891', 1, 0003),
( '978234567893', 1, 0001);
```

### Exercício 02:

Utilizando o banco de dados MySQL, executar os scripts SQL do exercício 01 para a definição do banco de dados do caso de uso da Livraria. Com as tabelas devidamente criadas e preenchidas com os valores, realizar o script em SQL DML/DQL que execute as seguintes expressões algébricas relacionais (responder com o print da tabela resultante no MySQL):

1.

$$\sigma_{CIDADE = "Atibaia" \wedge BAIRRO = "Centro"}(CLIENTES)$$

```
100
101 • SELECT * FROM Clientes WHERE cidade = 'Atibaia' AND bairro = 'Centro';
102
```

	CODCLIENTE	nome	endereço	bairro	cidade
▶	1	Jorge Almeida	R. da Saudade, 33	Centro	Atibaia
*	NULL	NULL	NULL	NULL	NULL

2.

$$\pi_{ENDERECO, BAIRRO}(\sigma_{CIDADE = "Atibaia"}(CLIENTES))$$

```
103
104 • SELECT endereço, bairro FROM Clientes WHERE cidade = 'Atibaia';
---
```

	endereço	bairro
▶	R. da Saudade, 33	Centro
	Av. São João, 230	Jd. Paulista

### Exercício 03:

Escrever as seguintes consultas em SQL e adicionar o print da tabela resultante, para o mesmo caso de uso da Livraria:

1. Quais livrarias não existem na cidade onde residem os clientes?

```

203
204 • SELECT C.CodCliente, C.Nome, C.Endereco, C.Bairro, C.Cidade
205 FROM CLIENTES C
206 LEFT JOIN LIVRARIAS L ON C.Cidade = L.Cidade
207 WHERE L.Cidade IS NULL;
208
209
210

```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	CodCliente	Nome	Endereco	Bairro	Cidade
	2	Roberto Alves	R. XV de Novembro, 200	Pq. das Américas	Bragança Paulista
	3	Juvenal Bueno	Av. Santos Dumont, 350	Centro	Joanópolis

2. Encontrar o conjunto de tuplas que contém os clientes cujos pedidos de venda foram efetuados.

```

157 /
158 • SELECT DISTINCT Clientes.CODCLIENTE, Clientes.nome, Clientes.endereco, Clientes.bairro, Clientes.cidade
159 FROM Clientes
160 INNER JOIN Pedidos ON Clientes.CODCLIENTE = Pedidos.CODCLIENTE;
161

```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	CODCLIENTE	nome	endereco	bairro	cidade
▶	1	Jorge Almeida	R. da Saudade, 33	Centro	Atibaia
	2	Roberto Alves	R. XV de Novembro, 200	Pq. das Américas	Bragança Paulista
	3	Juvenal Bueno	Av. Santos Dumont, 350	Centro	Joanópolis
	4	Marcos Rodrigues	Av. São João, 230	Jd. Paulista	Atibaia

3. Listar os livros que os clientes compraram.

```

161
162 • SELECT DISTINCT Titulos.ISBN, Titulos.titulo, Titulos.preco FROM Titulos
163 INNER JOIN Pedidos ON Titulos.ISBN = Pedidos.ISBN;

```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	ISBN	titulo	preco
▶	9781234567890	Computação e Animação Gráfica com Blender 2.80	89.00
	978234567893	Kotlin na Prática	85.00
	9782234567891	Introdução à Programação de Computadores	68.00
	978234567892	Desenvolvimento para Web	60.00

4. Listar todos os clientes que compraram os mesmos livros adquiridos pelo cliente Marcos Rodrigues (código 0004).

```

173
174 • SELECT DISTINCT Clientes.CODCLIENTE, Clientes.nome, Titulos.titulo
175 FROM Clientes
176 INNER JOIN Pedidos ON Clientes.CODCLIENTE = Pedidos.CODCLIENTE
177 INNER JOIN Titulos ON Pedidos.ISBN = Titulos.ISBN
178 WHERE Pedidos.ISBN IN (
179     SELECT ISBN
180 FROM Pedidos
181 WHERE CODCLIENTE = 0004
182 )
183

```

	CODCLIENTE	nome	titulo
▶	4	Marcos Rodrigues	Introdução à Programação de Computadores
	3	Juvenal Bueno	Introdução à Programação de Computadores
	2	Roberto Alves	Kotlin na Prática
	4	Marcos Rodrigues	Kotlin na Prática
	1	Jorge Almeida	Kotlin na Prática

- Existe também a opção de fazer essa consulta retirando o nome do cliente Marcos Rodrigues, deixando apenas os outros clientes que compraram o mesmo livro. Essa consulta é assim:

```

173
174 • SELECT DISTINCT Clientes.CODCLIENTE, Clientes.nome, Titulos.titulo
175 FROM Clientes
176 INNER JOIN Pedidos ON Clientes.CODCLIENTE = Pedidos.CODCLIENTE
177 INNER JOIN Titulos ON Pedidos.ISBN = Titulos.ISBN
178 WHERE Pedidos.ISBN IN (
179     SELECT ISBN
180 FROM Pedidos
181 WHERE CODCLIENTE = 0004
182 )
183 AND Clientes.CODCLIENTE != 0004;
184

```

	CODCLIENTE	nome	titulo
▶	3	Juvenal Bueno	Introdução à Programação de Computadores
	2	Roberto Alves	Kotlin na Prática
	1	Jorge Almeida	Kotlin na Prática