

BANCO DE DADOS

Trabalho – Relatório

| | |
|------------------|---------------------------------------|
| Curso: | Análise e Desenvolvimento de Sistemas |
| Aluno(a): | Gabriela Alves do Nascimento |
| RU: | 4737941 |

1. 1ª Etapa – Modelagem

Pontuação: 25 pontos.

Dadas as regras de negócio abaixo listadas, referentes ao estudo de caso de uma Clínica Médica, elabore o Modelo Entidade-Relacionamento (MER), isto é, o modelo conceitual.

O Modelo Entidade-Relacionamento (MER) deve contemplar os seguintes itens:

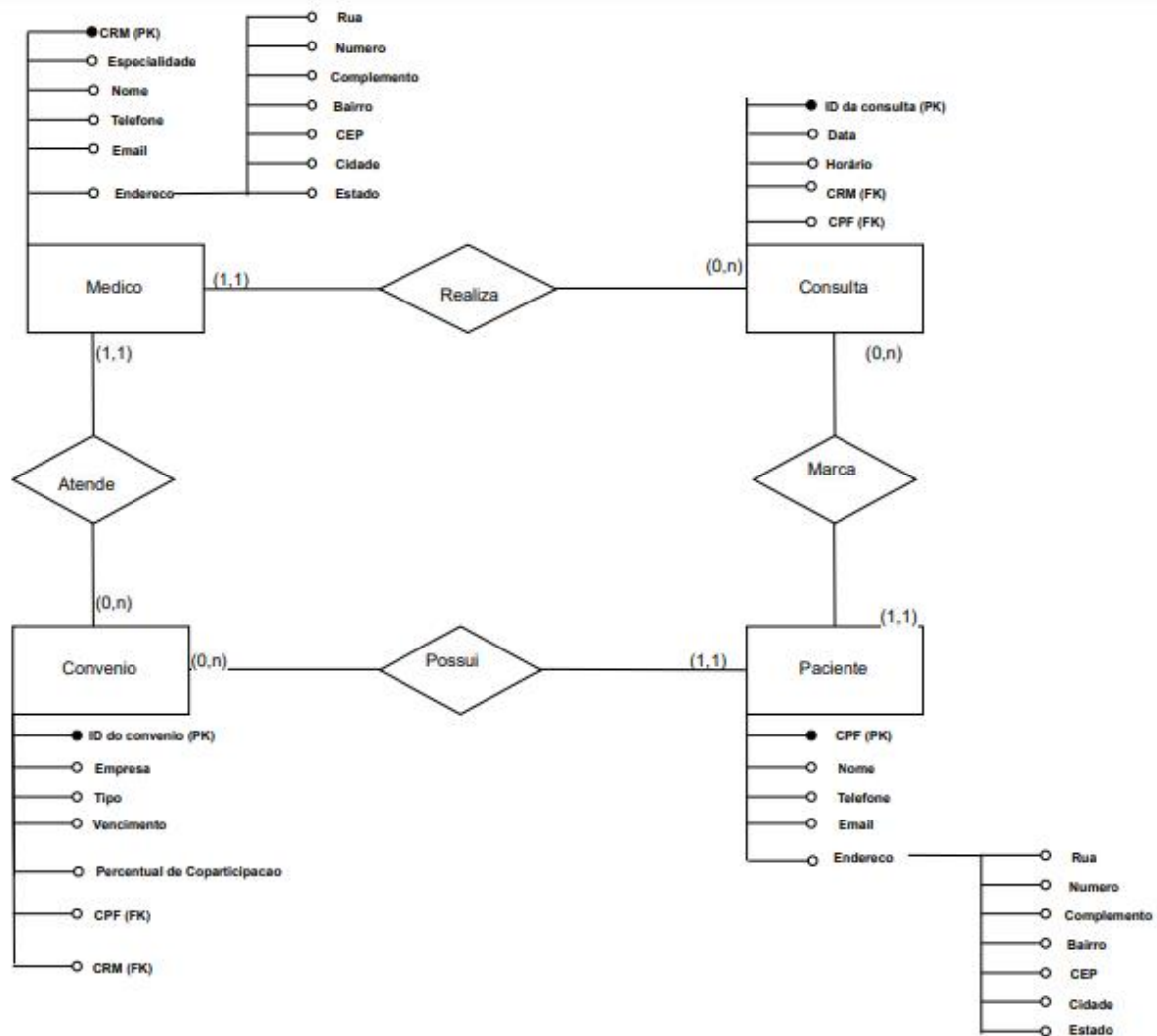
- Entidades;
- Atributos;
- Relacionamentos;
- Cardinalidades;
- Chaves primárias;
- Chaves estrangeiras.

Uma Clínica Médica necessita controlar os dados das consultas realizadas. Para isso, contratou um profissional de Banco de Dados, a fim de modelar o Banco de Dados que armazenará os dados das consultas.

As regras de negócio são:

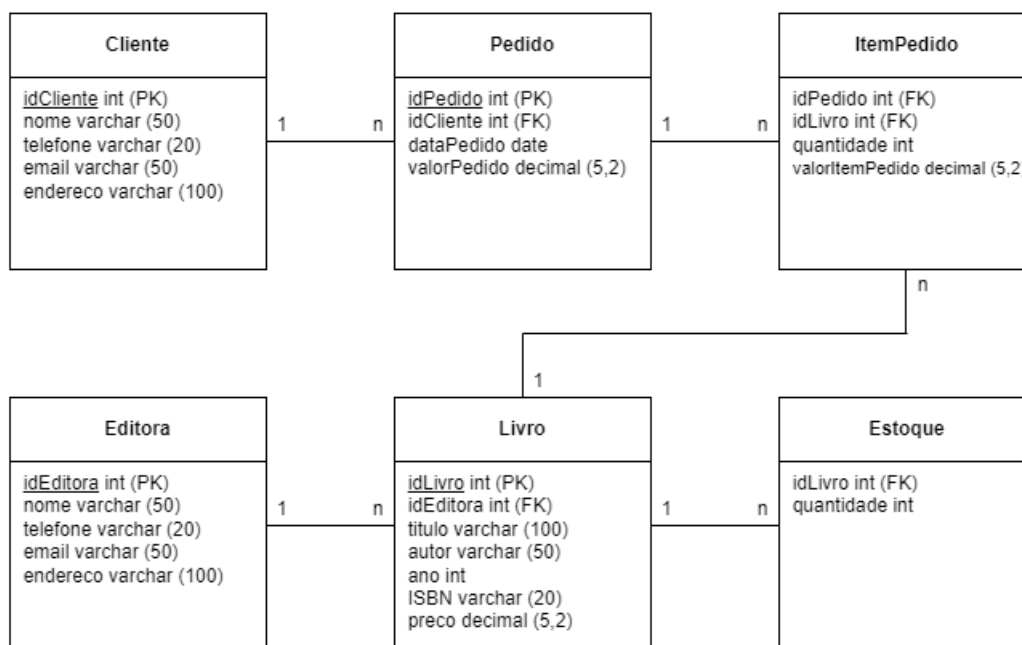
- Médico – Deverão ser armazenados os seguintes dados: CRM, especialidade, nome, telefone, e-mail e endereço, sendo o endereço composto por rua, número, complemento, bairro, CEP, cidade e estado;

- Consulta – Deverão ser armazenados os seguintes dados: identificação da consulta, data e horário;
- Paciente – Deverão ser armazenados os seguintes dados: CPF, nome, telefone, e-mail e endereço, sendo o endereço composto por rua, número, complemento, bairro, CEP, cidade e estado;
- Convênio – Deverão ser armazenados os seguintes dados: identificação do convênio, empresa, tipo, vencimento e percentual de coparticipação;
- Um médico pode realizar zero ou várias consultas, assim como zero ou várias consultas podem ser realizadas por um médico;
- Um médico pode atender zero ou vários convênios, assim como zero ou vários convênios podem ser atendidos por um médico;
- Um paciente pode marcar zero ou várias consultas, assim como zero ou várias consultas podem ser marcadas por um paciente;
- Um paciente pode possuir zero ou vários convênios, assim como zero ou vários convênios podem pertencer a um paciente.



2. 2ª Etapa – Implementação

Considere o seguinte Modelo Relacional (modelo lógico), referente ao estudo de caso de uma Livraria:



Com base no Modelo Relacional dado e utilizando a *Structured Query Language* (SQL), no MySQL Workbench, implemente o que se pede.

Observação: Para testar o Banco de Dados após a implementação, utilize os comandos contidos no arquivo “Trabalho – Populando o Banco de Dados” para popular as tabelas. Tal arquivo contém todos os comandos de inserção dos dados (fictícios) necessários para a realização dos testes.

Pontuação: 25 pontos.

1. Implemente um Banco de Dados chamado “Livraria”. Após, implemente as tabelas, conforme o Modelo Relacional dado, observando as chaves primárias e as chaves estrangeiras. Todos os campos, de todas as tabelas, não podem ser nulos (*not null*).

create database Livraria;

use Livraria;

CREATE TABLE Cliente (

idCliente int auto_increment NOT NULL primary key,

nome VARCHAR(50) NOT NULL,

```
telefone VARCHAR(20) NOT NULL,  
email VARCHAR(50) NOT NULL,  
endereco VARCHAR(100) NOT NULL  
);  
CREATE TABLE Pedido (  
idPedido int NOT NULL primary key,  
dataPedido date NOT NULL,  
valorPedido decimal (5,2) NOT NULL,  
idCliente int NOT NULL,  
foreign key (idCliente) references Cliente(idCliente));
```

```
CREATE TABLE Editora (  
idEditora int NOT NULL primary key,  
nome VARCHAR(50) NOT NULL,  
telefone VARCHAR(20) NOT NULL,  
email VARCHAR(50) NOT NULL,  
endereco VARCHAR(100) NOT NULL);
```

```
CREATE TABLE Livro (  
idLivro int NOT NULL primary key,  
idEditora int NOT NULL,  
titulo VARCHAR(100) NOT NULL,  
autor VARCHAR(50) NOT NULL,  
ano int NOT NULL,  
ISBN VARCHAR(20) NOT NULL,  
preco decimal (5,2) NOT NULL,  
foreign key (idEditora) references Editora(idEditora));
```

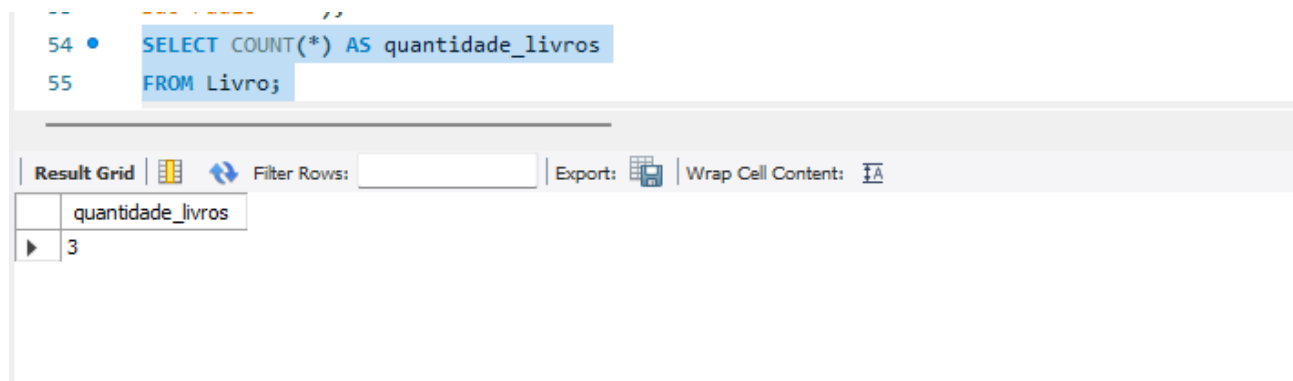
```
CREATE TABLE Estoque (  
idLivro int NOT NULL,  
quantidade int NOT NULL,  
foreign key (idLivro) references Livro(idLivro));
```

```
CREATE TABLE ItemPedido (  
idLivro int NOT NULL,  
idPedido int NOT NULL,  
quantidade int NOT NULL,  
valorItemPedido decimal (5,2) NOT NULL,  
foreign key (idLivro) references Livro(idLivro),  
foreign key (idPedido) references Pedido(idPedido));
```

Pontuação: 10 pontos.

2. Implemente uma consulta para listar o quantitativo de livros cadastrados, independentemente da editora.

```
SELECT COUNT(*) AS quantidade_livros FROM Livro;
```



Pontuação: 10 pontos.

3. Implemente uma consulta para listar o nome dos clientes cadastrados. A listagem deve ser mostrada em ordem crescente.

```
select nome from Cliente order by nome ASC;
```

```
64 • select nome from Cliente order by nome ASC;
65
```

| nome |
|----------------|
| Gabriela Alves |
| Joana Santos |
| João Silva |

Pontuação: 10 pontos.

4. Implemente uma consulta para listar o nome de todas as editoras e os títulos de seus respectivos livros. A listagem deve ser mostrada em ordem decrescente pelo nome das editoras.

```
SELECT e.nome AS editora_nome, l.titulo AS livro_titulo  
FROM Editora e  
JOIN Livro l ON e.idEditora = l.idEditora  
ORDER BY e.nome DESC;
```

```
65 • SELECT e.nome AS editora_nome, l.titulo AS livro_titulo
66 FROM Editora e
67 JOIN Livro l ON e.idEditora = l.idEditora
68 ORDER BY e.nome DESC;
```

| editora_nome | livro_titulo |
|--|-------------------------|
| Editora Schwarcz S.A. | O HERDEIRO INADEQUADO |
| Editora Intrínseca | VOU TE RECEITAR UM GATO |
| Distribuidora Record de Serviços de Imprensa S.A | É assim que acaba |

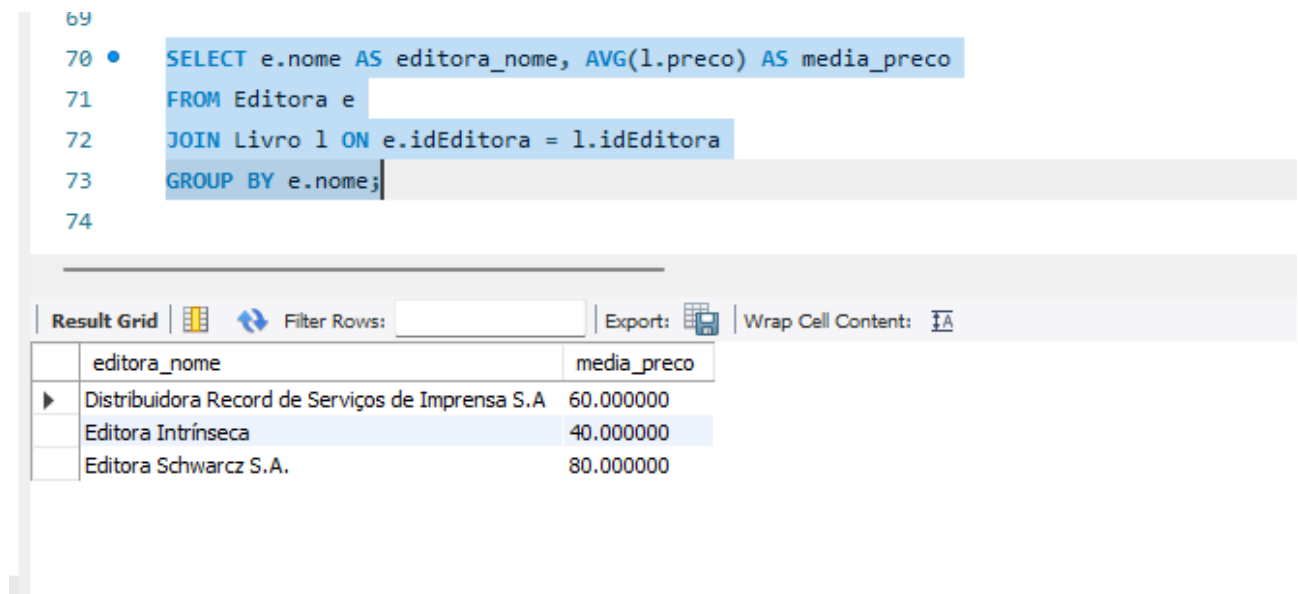
Result 9 x

Pontuação: 10 pontos.

5. Implemente uma consulta para listar o nome das editoras e a média de preço de seus respectivos livros. Para isso, utilize o comando *group by*.

```
SELECT e.nome AS editora_nome, AVG(l.preco) AS media_preco
```

```
FROM Editora e
JOIN Livro l ON e.idEditora = l.idEditora
GROUP BY e.nome;
```



```
69
70 • SELECT e.nome AS editora_nome, AVG(l.preco) AS media_preco
71 FROM Editora e
72 JOIN Livro l ON e.idEditora = l.idEditora
73 GROUP BY e.nome;
74
```

| editora_nome | media_preco |
|--|-------------|
| Distribuidora Record de Serviços de Imprensa S.A | 60.000000 |
| Editora Intrínseca | 40.000000 |
| Editora Schwarcz S.A. | 80.000000 |

Pontuação: 10 pontos.

6. Implemente uma consulta para listar o nome de todos os clientes e a quantidade de livros comprados pelos mesmos. Para isso, utilize o comando *group by*.

```
SELECT c.nome AS cliente_nome, COUNT(co.idPedido) AS quantidade_livros
FROM Cliente c
LEFT JOIN Pedido co ON c.idCliente = co.idCliente
GROUP BY c.nome;
```



```
29 • SELECT c.nome AS cliente_nome, COUNT(co.idPedido) AS quantidade_livros
30 FROM Cliente c
31 LEFT JOIN Pedido co ON c.idCliente = co.idCliente
32 GROUP BY c.nome;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [↻](#)

| | cliente_nome | quantidade_livros |
|---|----------------|-------------------|
| ▶ | Gabriela Alves | 2 |
| | João Silva | 1 |
| | Joana Santos | 2 |