

BANCO DE DADOS

Trabalho – Relatório

Curso:	CST em Análise e Desenvolvimento de Sistemas
Aluno(a):	José Antônio Garcia Lehnemann Filho
RU:	4695081

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:

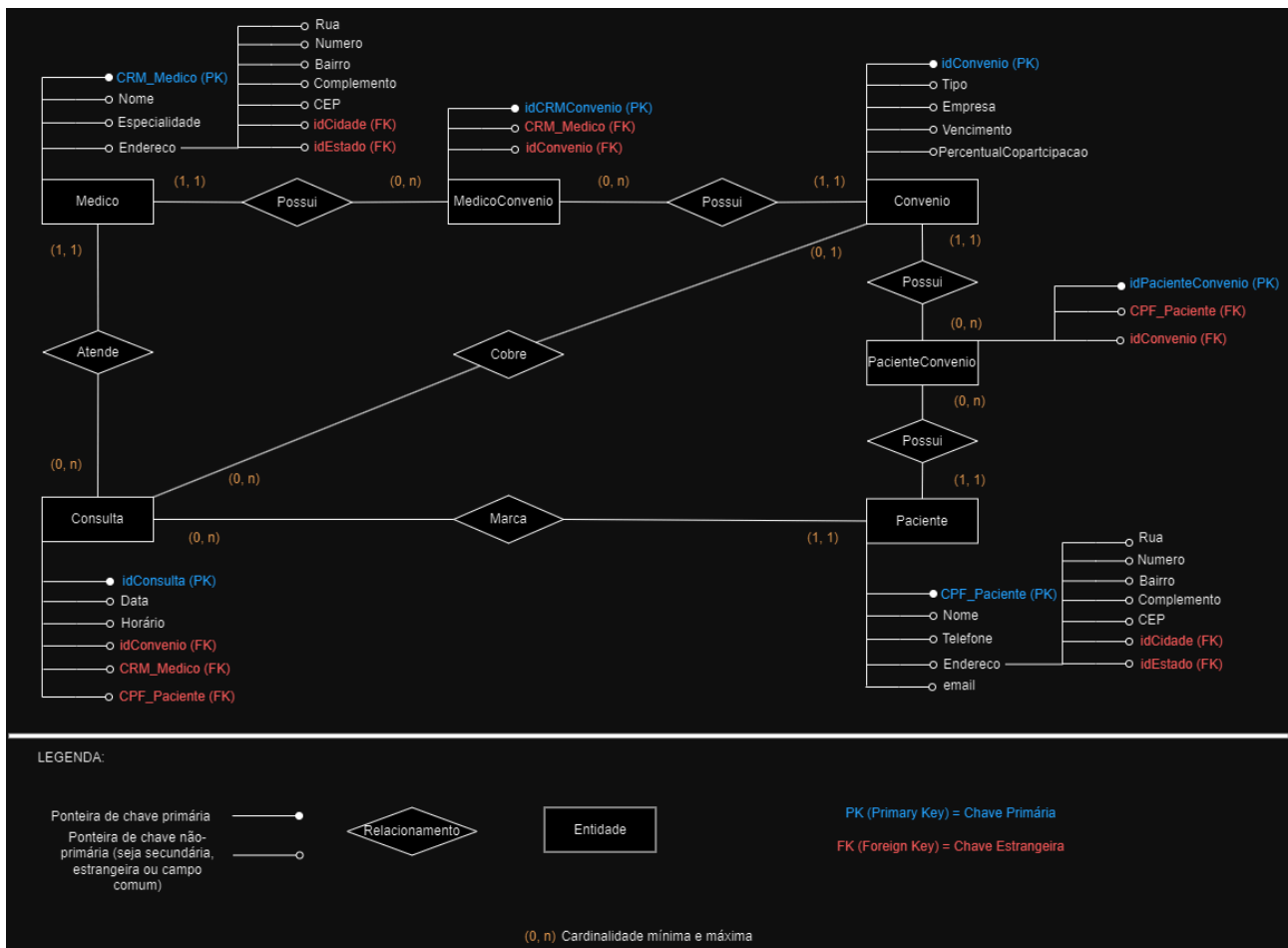
- Entidades;
- Atributos;
- Relacionamentos;
- Cardinalidades;
- Chaves primárias;
- Chaves estrangeiras.

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:

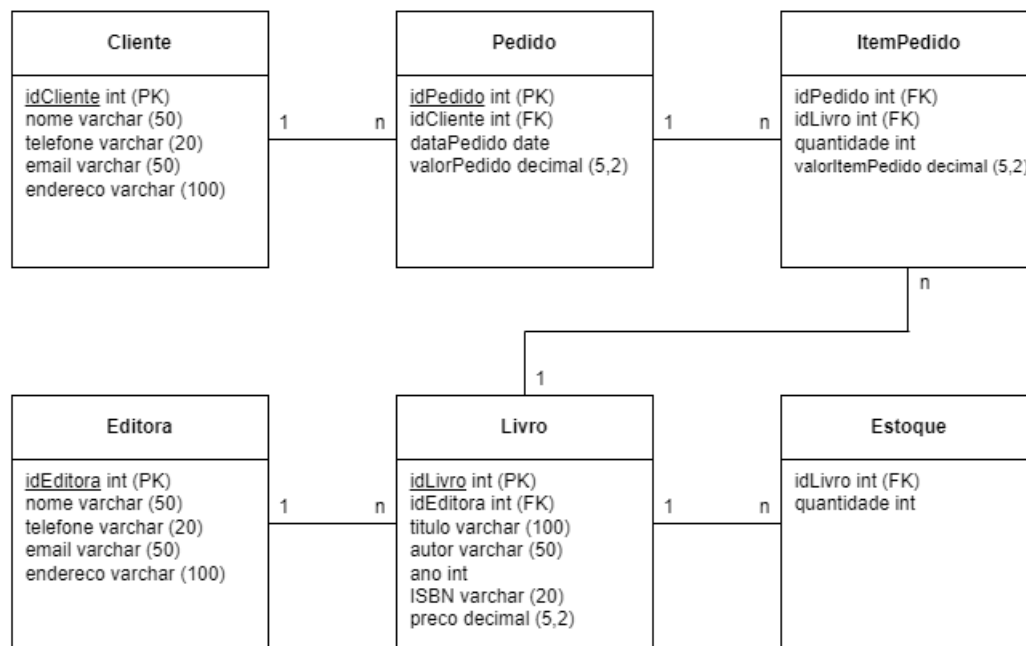
- Médico – Deverão ser armazenados os seguintes dados: CRM, especialidade, nome, telefone, e-mail e endereço, sendo o endereço composto por rua, número, complemento, bairro, CEP, cidade e estado;

-
- 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 not null auto_increment,  
nome varchar(50) not null,  
telefone varchar(20) not null,  
email varchar(50) not null,  
endereco varchar(100) not null,  
constraint pk_idCliente primary key (idCliente)  
);
```

```
create table Pedido(  
    idPedido int not null auto_increment,  
    idCliente int not null,  
    dataPedido date not null,  
    valorPedido decimal(5,2) not null,  
    constraint pkPedido primary key (idPedido),  
    constraint fk_idCliente foreign key (idCliente) references Cliente (idCliente)  
on delete no action on update cascade  
);
```

```
create table Editora(  
    idEditora int not null auto_increment,  
    nome varchar(50) not null,  
    telefone varchar(50) not null,  
    email varchar(50) not null,  
    endereco varchar(100) not null,  
    constraint pk_idEditora primary key (idEditora)  
);
```

```
create table Livro(  
    idLivro int not null auto_increment,  
    idEditora int not null,  
    titulo varchar(100) not null,  
    autor varchar(50) not null,
```

```
ano int not null,  
ISBN varchar(50),  
preco decimal (5,2),  
constraint pk_idLivro primary key (idLivro),  
constraint fk_idEditora foreign key (idEditora) references Editora (idEditora)  
on delete no action on update cascade  
);
```

```
create table ItemPedido(  
    idPedido int not null,  
    idLivro int not null,  
    quantidade int not null,  
    valorItemPedido decimal(5,2),  
    constraint fk_idPedido_ItemPedido foreign key (idPedido) references Pedido  
(idPedido) on delete no action on update cascade,  
    constraint fk_idLivro_ItemPedido foreign key (idLivro) references Livro  
(idLivro) on delete no action on update cascade  
);
```

```
create table Estoque(  
    idLivro int not null,  
    quantidade int not null,  
    constraint fk_idLivro_Estoque foreign key (idLivro) references Livro (idLivro)  
on delete no action on update cascade  
);
```

Pontuação: 10 pontos.

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

```
select count(idLivro) as 'Quantidade de Livros Cadastrados' from Livro;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
#	Quantidade de Livros Cadastrados			
1	15			

Pontuação: 10 pontos.

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

select nome as 'Nome do Cliente' from Cliente;

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
#	Nome do Cliente			
1	Alice de Souza			
2	Mario Vicente			
3	Maria Helena Mantovani			
4	Vitor Martins			
5	Nicole Amanda de Jesus			
6	Luciano Tucolo			
7	Paula Roberta Vitorino			
8	Guilherme Koeriche			
9	Beatriz Leopoldina			
10	Lucas Cochuelo			

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 Editora.nome as 'Editora', Livro.titulo as 'Título do Livro'
from Editora left join Livro
on Editora.idEditora = Livro.idEditora order by Editora.nome desc;

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
#	Editora	Título do Livro			
1	Sexta dos Estudos	Matemática Descomplicada			
2	Sexta dos Estudos	Português para Estrangeiros			
3	Sexta dos Estudos	Estatística é para Todos			
4	Mais Informática	Aprendendo Python em 24 Horas			
5	Mais Informática	Banco de Dados: Aprenda de Form...			
6	Mais Informática	Java para Que Te Quero			
7	Companhia da Leitura	Minha Faculdade Vai Me Enlouquecer			
8	Companhia da Leitura	Controlando as Emoções			
9	Companhia da Leitura	Brasil Brasileiro			
10	Ciência da Informação	Big Data: Conhecimentos Essenciais			
11	Ciência da Informação	Ciência de Dados: O Futuro			
12	Ciência da Informação	Inteligência Artificial Aplicada a Dados			
13	Arco da Velha	Se Eu Voltasse no Passado			
14	Arco da Velha	Penso, Logo Existo			
15	Arco da Velha	Lugares para Viajar Sozinho			

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 Editora.nome as 'Editora', round(avg(Livro.Preco),2) as 'Média de preço dos livros'
from Editora inner join Livro
on Editora.idEditora = Livro.idEditora
group by Editora.nome;
```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
#	Editora	Média de preço dos livros			
1	Companhia da Leitura	38.57			
2	Arco da Velha	56.23			
3	Mais Informática	106.57			
4	Sexta dos Estudos	76.57			
5	Ciência da Informação	147.23			

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 Cliente.nome as 'Nome do Cliente', sum(ItemPedido.quantidade) as  
'Quantidade de livros comprados pelo cliente'  
from Cliente left join Pedido on Cliente.idCliente = Pedido.idCliente  
inner join ItemPedido on Pedido.idPedido = ItemPedido.idPedido  
group by Cliente.nome;
```

#	Nome do Cliente	Quantidade de livros comprados pelo cliente
1	Alice de Souza	1
2	Mario Vicente	1
3	Maria Helena Mantovani	4
4	Vitor Martins	1
5	Nicole Amanda de Jesus	1
6	Luciano Tucolo	4
7	Paula Roberta Vitorino	2
8	Guilherme Koeriche	1
9	Beatriz Leopoldina	2
10	Lucas Cochuelo	2