

BANCO DE DADOS

Trabalho – Relatório

Curso:	Análise e desenvolvimento de sistemas (ADS)
Aluno(a):	Pedro Henrique Silva de Farias
RU:	4845775

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:

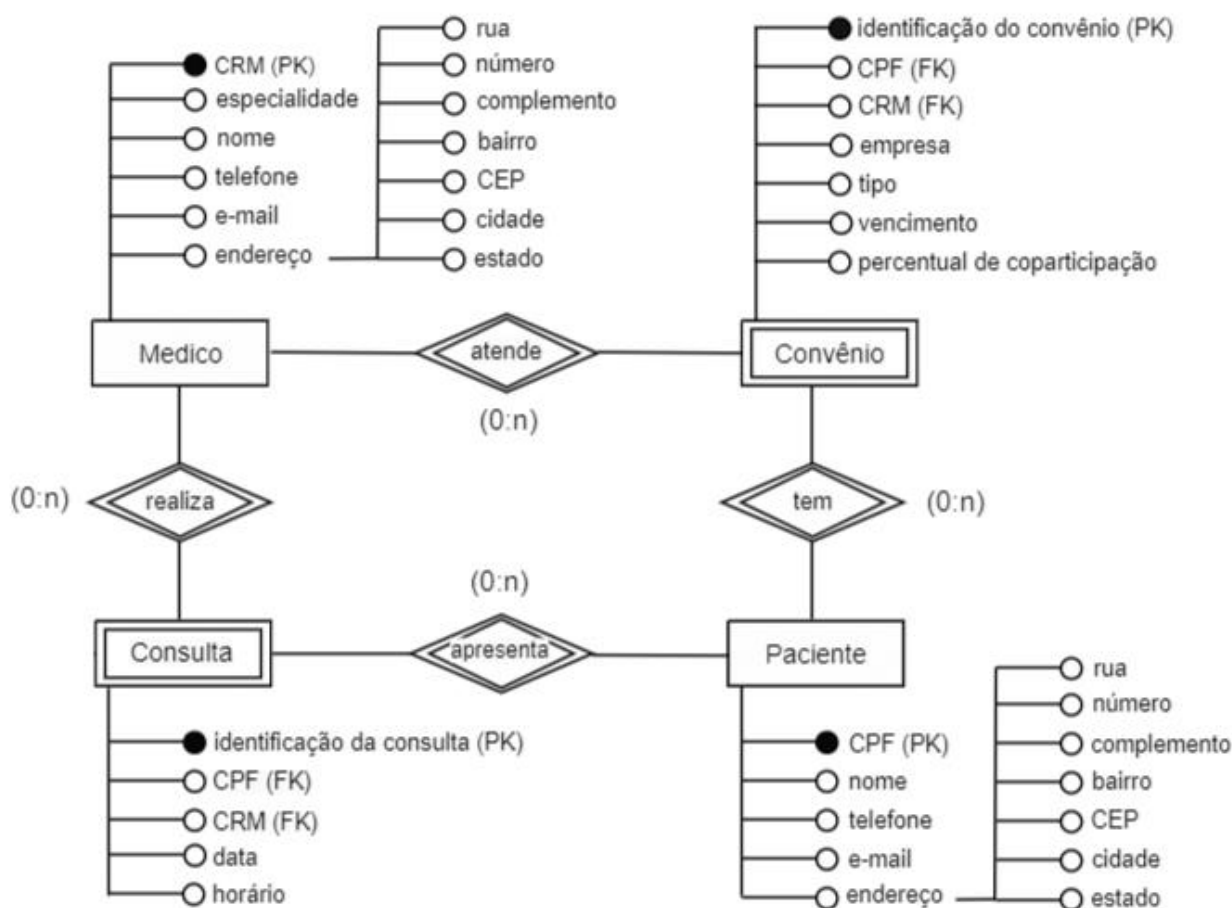
- Entidades;
- Atributos;
- Relacionamentos;
- Cardinalidades;
- Chaves primárias;
- Chaves estrangeiras.

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:

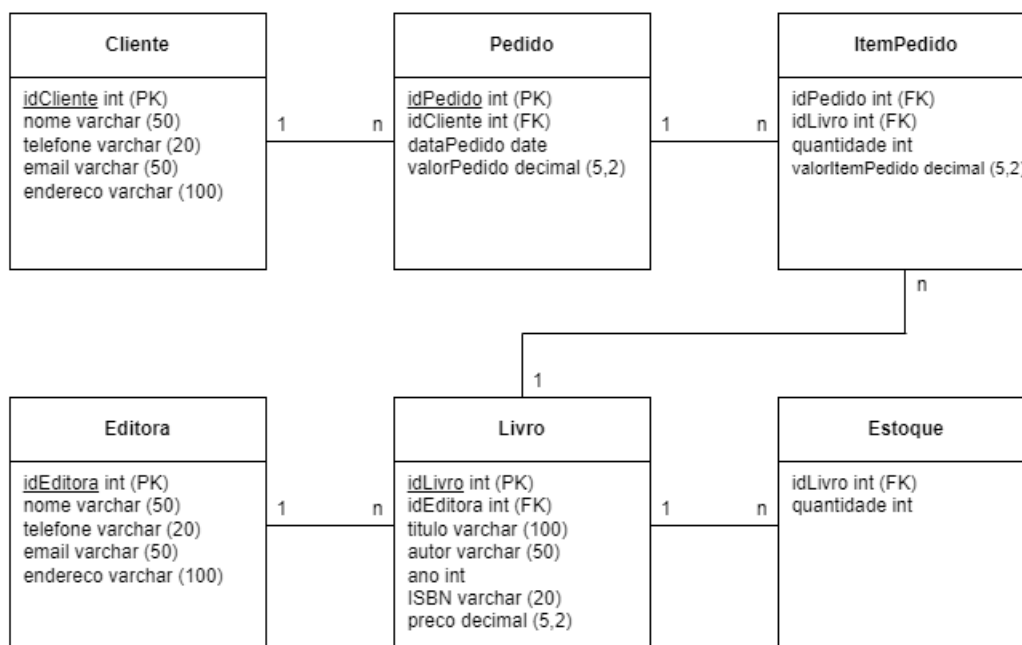
- Médico – Deverão ser armazenados os seguintes dados: CRM, especialidade, nome, telefone, e-mail e endereço, sendo o endereço composto por rua, número, complemento, bairro, CEP, cidade e estado;

- 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 primary key not null,  
    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 primary key not null,  
    idCliente int not null,  
    dataPedido date not null,  
    valorPedido decimal(5,2) not null,  
    foreign key (idCliente) references Cliente(idCliente)  
);  
  
create table Editora (  
    idEditora int primary key not null,  
    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 primary key not null,  
    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)
```

);

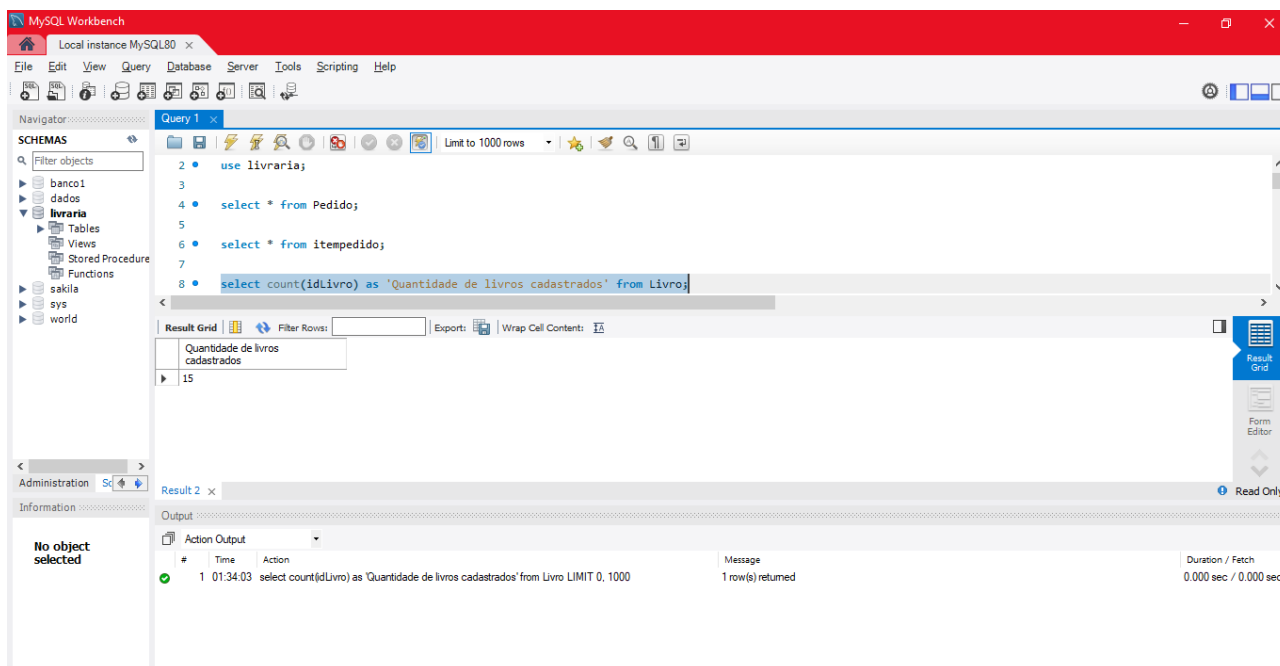
```
create table ItemPedido (  
    idPedido int not null,  
    idLivro int not null,  
    quantidade int not null,  
    valorItemPedido decimal (5,2) not null,  
    foreign key (idPedido) references Pedido(idPedido),  
    foreign key (idLivro) references Livro(idLivro)  
);
```

```
create table Estoque (  
    idLivro int not null,  
    quantidade int not null,  
    foreign key (idLivro) references Livro(idLivro)  
);
```

Pontuação: 10 pontos.

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

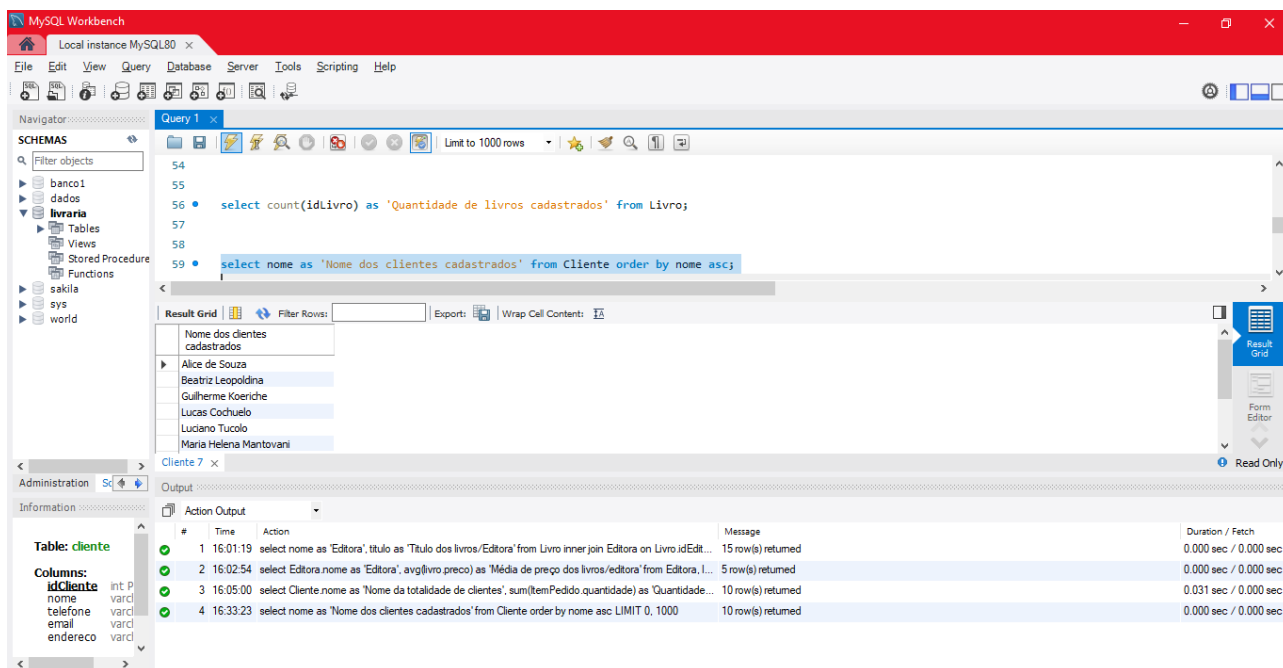
```
select count(idLivro) as 'Quantidade de livros cadastrados' from Livro;
```



Pontuação: 10 pontos.

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

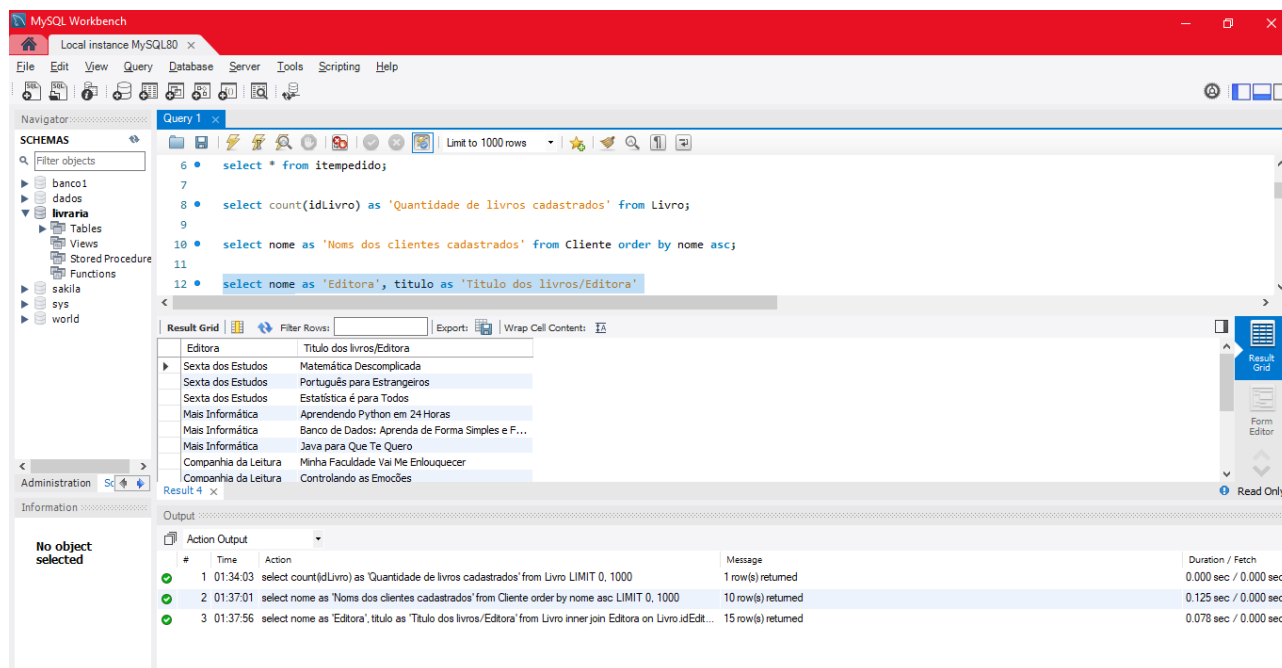
```
select nome as 'Nome dos clientes cadastrados' from Cliente order by nome asc;
```



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.

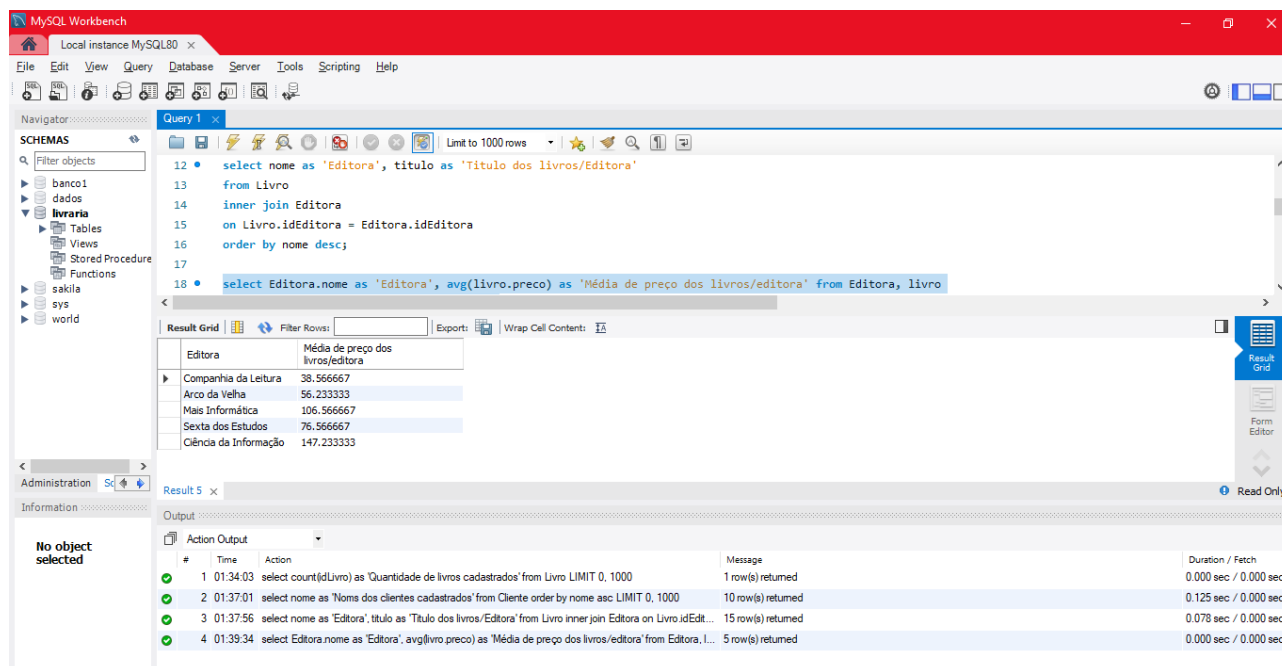
```
select nome as 'Editora', titulo as 'Titulo dos livros/Editora'
from Livro
inner join Editora
on Livro.idEditora = Editora.idEditora
order by nome desc;
```

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', avg(livro.preco) as 'Média de preço dos livros/editora'
from Editora, livro
where Editora.idEditora = livro.idEditora
group by nome;
```



The screenshot shows the MySQL Workbench interface. The 'Query' tab is active, displaying a SQL query that joins the 'Editora' and 'Livro' tables to calculate the average price of books for each publisher. The 'Result Grid' shows the output of the query, listing publishers and their average book prices. The 'Action Output' tab at the bottom shows the execution log of the queries.

```
12 • select nome as 'Editora', titulo as 'Título dos livros/Editora'
13 from Livro
14 inner join Editora
15 on Livro.idEditora = Editora.idEditora
16 order by nome desc;
17
18 • select Editora.nome as 'Editora', avg(livro.preco) as 'Média de preço dos livros/editora' from Editora, livro
```

Editora	Média de preço dos livros/editora
Companhia da Leitura	38.566667
Arco da Velha	56.233333
Mais Informática	106.566667
Sexta dos Estudos	76.566667
Ciência da Informação	147.233333

#	Time	Action	Message	Duration / Fetch
1	01:34:03	select count(idLivro) as 'Quantidade de livros cadastrados' from Livro LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
2	01:37:01	select nome as 'Noms dos clientes cadastrados' from Cliente order by nome asc LIMIT 0, 1000	10 row(s) returned	0.125 sec / 0.000 sec
3	01:37:56	select nome as 'Editora', titulo as 'Título dos livros/Editora' from Livro inner join Editora on Livro.idEdit...	15 row(s) returned	0.078 sec / 0.000 sec
4	01:39:34	select Editora.nome as 'Editora', avg(livro.preco) as 'Média de preço dos livros/editora' from Editora, L...	5 row(s) returned	0.000 sec / 0.000 sec

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 da totalidade de clientes', sum(ItemPedido.quantidade) as
'Quantidade de livros comprados/cliente' from Cliente, ItemPedido
where Cliente.idCliente = ItemPedido.idPedido
group by Cliente.idCliente
having      count(quantidade)      =      count(ItemPedido.valorItemPedido)      and
count(ItemPedido.valorItemPedido) = count(ItemPedido.idPedido);
```

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

banco1

dados

livraria

Tables

Views

Stored Procedure

Functions

sakila

sys

world

Query 1

Limit to 1000 rows

```
21
22
23 • select Cliente.nome as 'Nome da totalidade de clientes', sum(ItemPedido.quantidade) as 'Quantidade de livros comprados/cliente' from Cliente, ItemPedido
24 where Cliente.idCliente = ItemPedido.idPedido
25 group by Cliente.idCliente
26 having count(quantidade) = count(ItemPedido.valorItemPedido) and count(ItemPedido.valorItemPedido) = count(ItemPedido.idPedido);
27
```

Result Grid

Nome da totalidade de clientes	Quantidade de livros comprados/cliente
Alice de Souza	1
Mario Vicente	1
Maria Helena Mantovani	3
Vitor Martins	1
Nicole Amanda de Jesus	1
Luciano Tucolo	2
Paula Roberta Vitorino	2

Administration

Information

No object selected

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	01:34:03	select count(idLivro) as 'Quantidade de livros cadastrados' from Livro LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
2	01:37:01	select nome as 'Noms dos clientes cadastrados' from Cliente order by nome asc LIMIT 0, 1000	10 row(s) returned	0.125 sec / 0.000 sec
3	01:37:56	select nome as 'Editora', titulo as 'Titulo dos livros/Editora' from Livro inner join Editora on Livro.idEdit...	15 row(s) returned	0.078 sec / 0.000 sec
4	01:39:34	select Editora.nome as 'Editora', avg(livro.preco) as 'Média de preço dos livros/editora' from Editora, L...	5 row(s) returned	0.000 sec / 0.000 sec
5	01:39:50	select Cliente.nome as 'Nome da totalidade de clientes', sum(ItemPedido.quantidade) as 'Quantidade...	10 row(s) returned	0.047 sec / 0.000 sec