

BANCO DE DADOS

Trabalho – Relatório

Curso:	Engenharia de software
Aluno(a):	Jefferson Alves Ferreira
RU:	4826463

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:

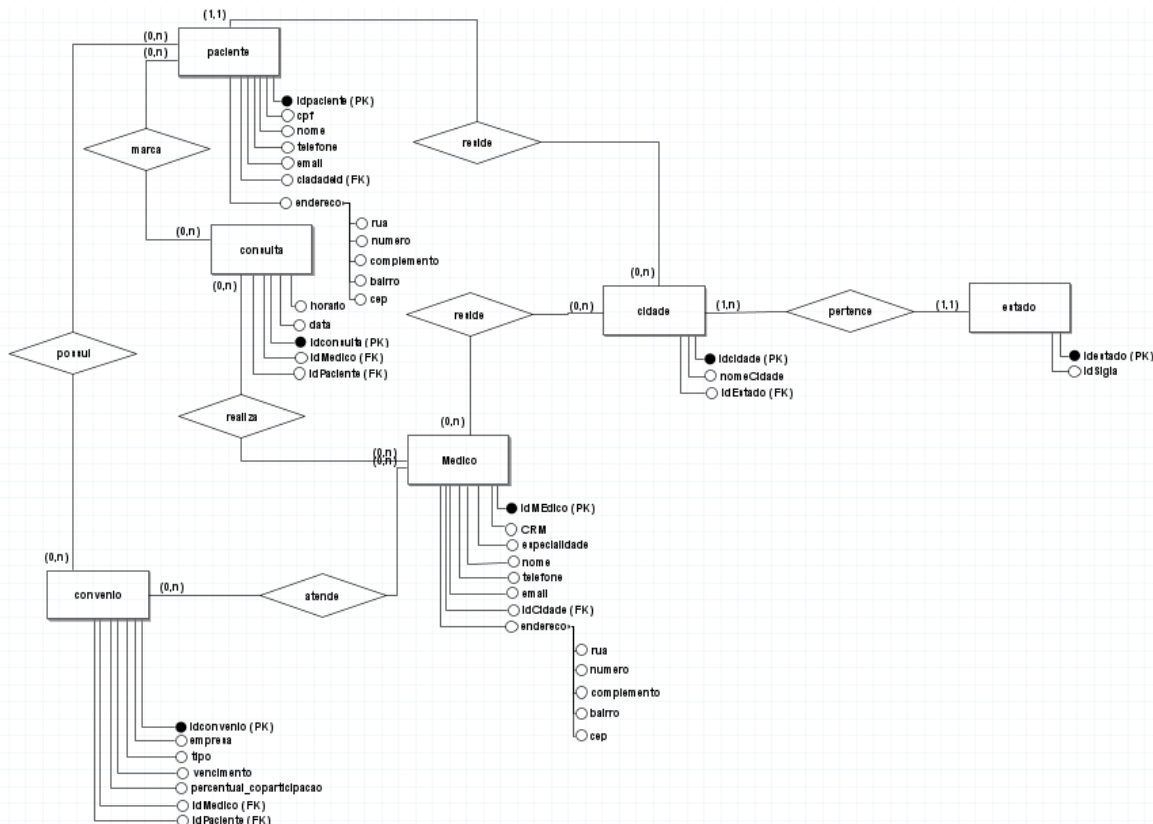
- Entidades;
- Atributos;
- Relacionamentos;
- Cardinalidades;
- Chaves primárias;
- Chaves estrangeiras.

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:

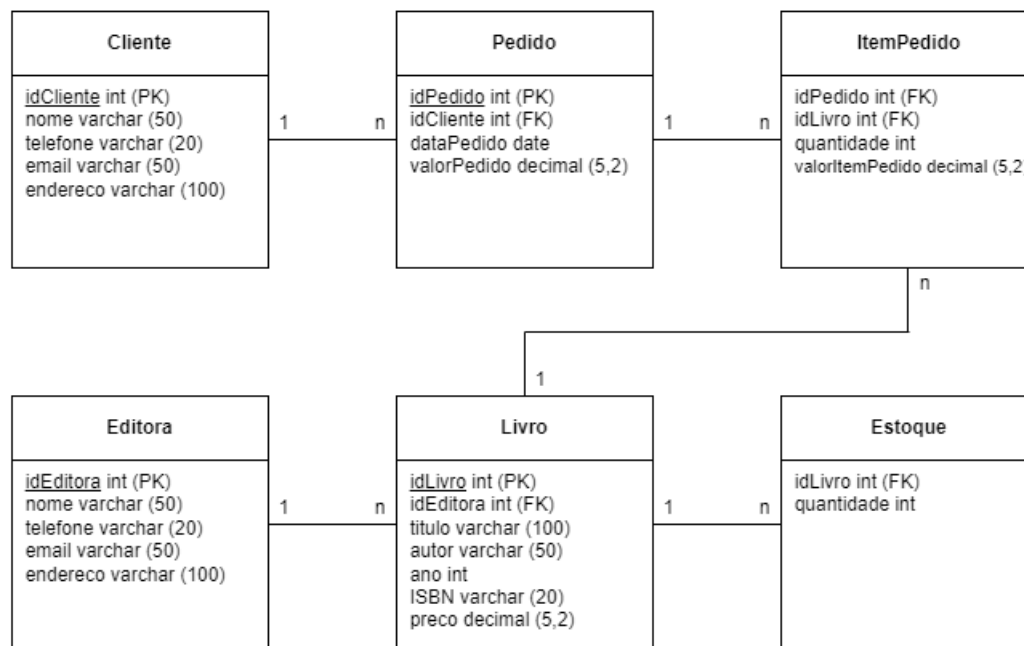
- Médico – Deverão ser armazenados os seguintes dados: CRM, especialidade, nome, telefone, e-mail e endereço, sendo o endereço composto por rua, número, complemento, bairro, CEP, cidade e estado;

- 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 AUTO_INCREMENT,  
    nome VARCHAR(50) NOT NULL,  
    telefone VARCHAR(20) NOT NULL,  
    email VARCHAR(50) NOT NULL,  
    endereco VARCHAR(100) NOT NULL  
) DEFAULT CHARACTER SET = utf8;
```

```
CREATE TABLE pedido (  
    idPedido INT PRIMARY KEY AUTO_INCREMENT,  
    dataPedido DATE NOT NULL,  
    valorPedido DECIMAL(10, 2) NOT NULL,  
    idCliente INT NOT NULL,  
    FOREIGN KEY (idCliente) REFERENCES Cliente(idCliente)  
) DEFAULT CHARACTER SET = utf8;
```

```
CREATE TABLE itemPedido (  
    idItemPedido INT PRIMARY KEY AUTO_INCREMENT,  
    quantidade INT NOT NULL,  
    valorItemPedido DECIMAL(10, 2) NOT NULL,  
    idPedido INT NOT NULL,  
    idLivro INT NOT NULL,  
    FOREIGN KEY (idPedido) REFERENCES pedido(idPedido),  
    FOREIGN KEY (idLivro) REFERENCES Livro(idLivro)  
) DEFAULT CHARACTER SET = utf8;
```

```
CREATE TABLE Editora (  
    idEditora INT PRIMARY KEY AUTO_INCREMENT,  
    nome VARCHAR(50) NOT NULL,  
    telefone VARCHAR(20) NOT NULL,  
    email VARCHAR(50) NOT NULL,
```

```
endereco VARCHAR(100) NOT NULL
) DEFAULT CHARACTER SET = utf8;

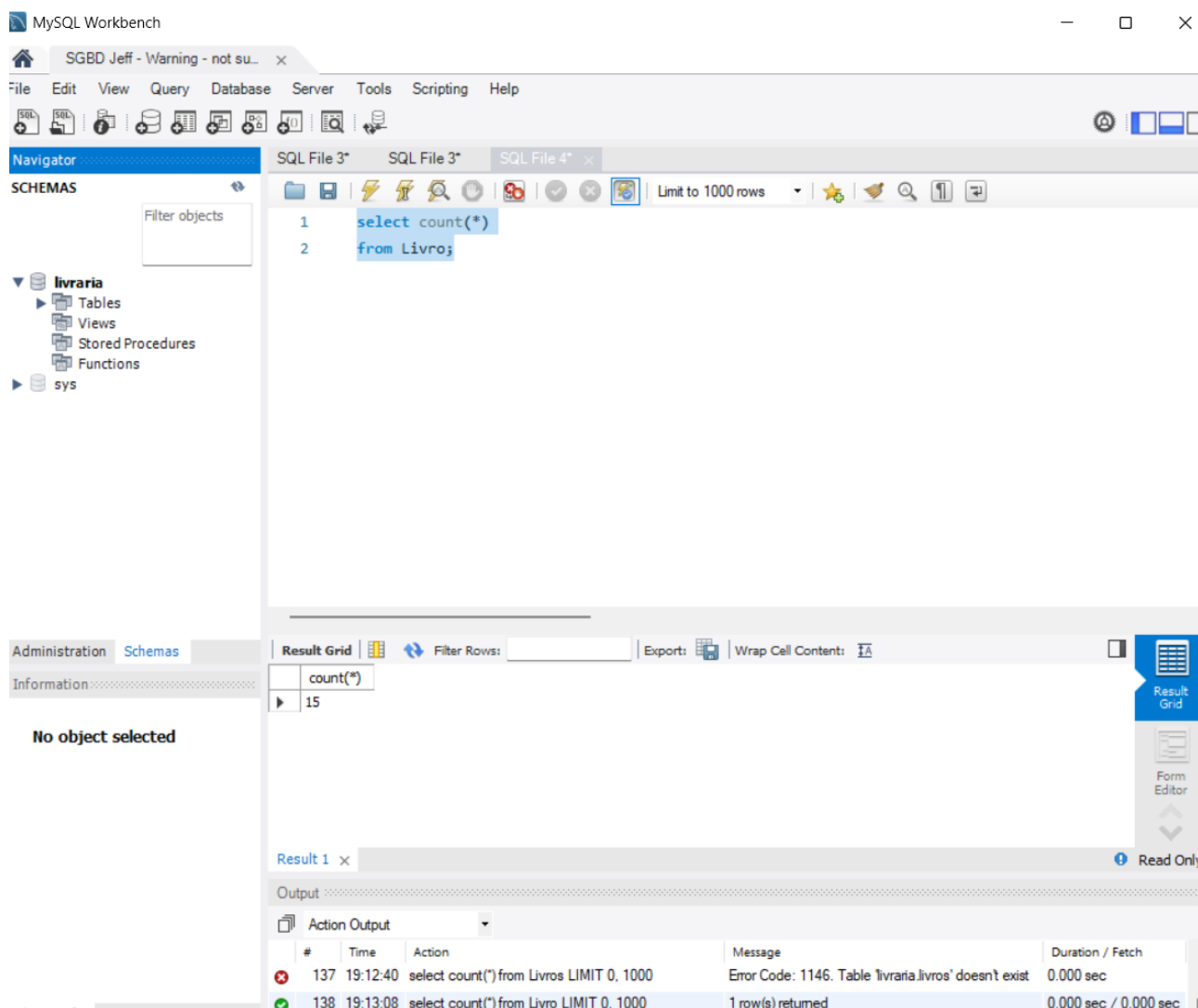
CREATE TABLE Livro (
    idLivro INT PRIMARY KEY AUTO_INCREMENT,
    titulo VARCHAR(100) NOT NULL,
    autor VARCHAR(50) NOT NULL,
    ano INT NOT NULL,
    isbn VARCHAR(20) NOT NULL,
    preco DECIMAL(10, 2) NOT NULL,
    idEditora INT NOT NULL,
    FOREIGN KEY (idEditora) REFERENCES Editora(idEditora)
) DEFAULT CHARACTER SET = utf8;

CREATE TABLE Estoque (
    idEstoque INT PRIMARY KEY AUTO_INCREMENT,
    quantidade INT NOT NULL,
    idLivro INT NOT NULL,
    FOREIGN KEY (idLivro) REFERENCES Livro(idLivro)
) DEFAULT CHARACTER SET = utf8;
```

Pontuação: 10 pontos.

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

```
select count(*)
from Livro;
```



The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 select count(*)
2 from Livro;
```

The left sidebar shows the Schemas tree with 'livraria' expanded, containing 'Tables', 'Views', 'Stored Procedures', and 'Functions'. The 'sys' schema is also visible.

The bottom panel shows the 'Result Grid' with one row:

count(*)
15

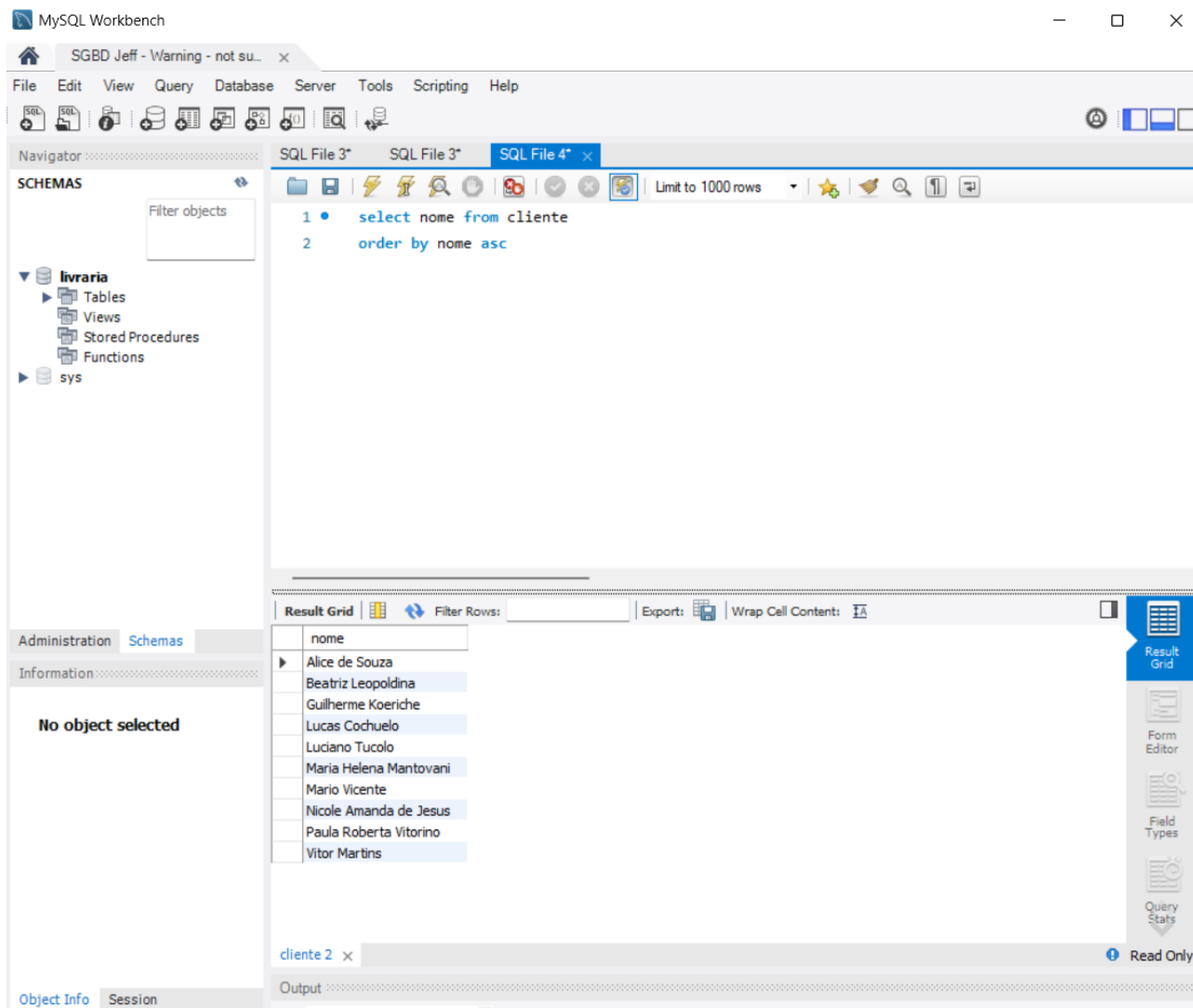
The 'Output' panel shows the 'Action Output' table:

#	Time	Action	Message	Duration / Fetch
137	19:12:40	select count(*) from Livros LIMIT 0, 1000	Error Code: 1146. Table 'livraria.livros' doesn't exist	0.000 sec
138	19:13:08	select count(*) from Livro LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

Pontuação: 10 pontos.

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

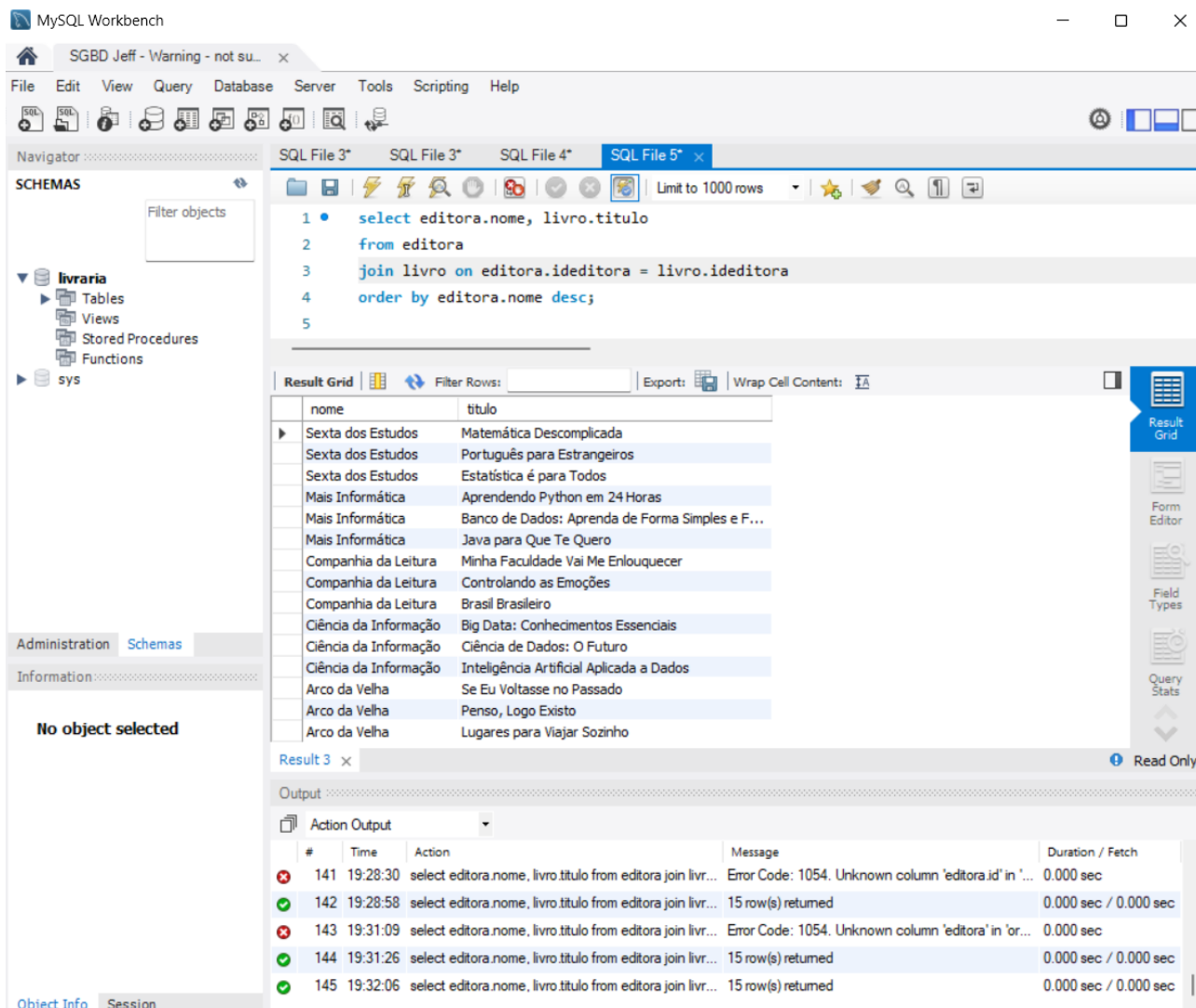
```
select nome from cliente
order by nome asc;
```



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, livro.titulo
from editora
join livro on editora.ideditora = livro.ideditora
order by editora.nome desc;
```



The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 select editora.nome, livro.titulo
2 from editora
3 join livro on editora.ideditora = livro.ideditora
4 order by editora.nome desc;
```

The Result Grid displays the following data:

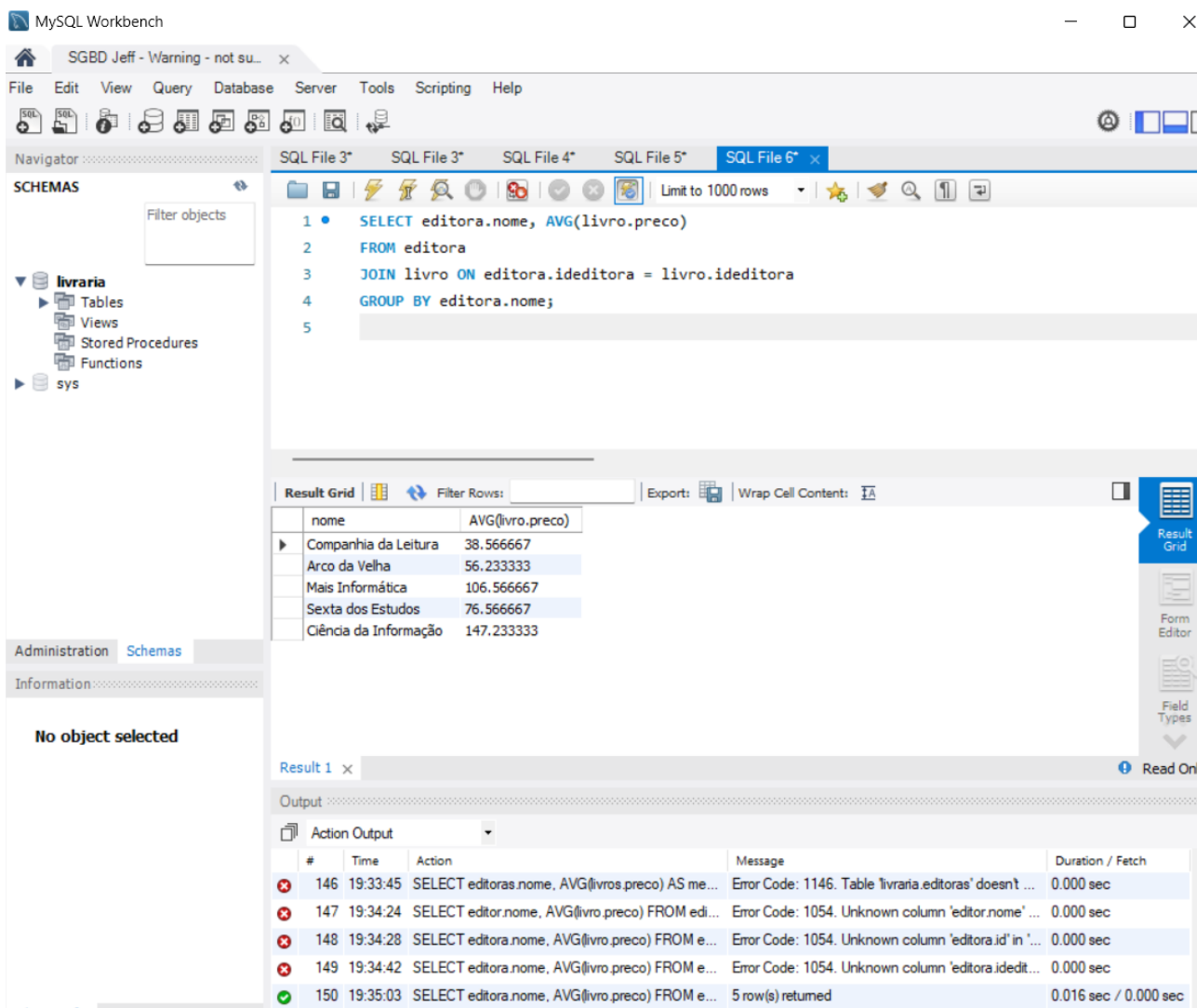
nome	titulo
Sexta dos Estudos	Matemática Descomplicada
Sexta dos Estudos	Português para Estrangeiros
Sexta dos Estudos	Estatística é para Todos
Mais Informática	Aprendendo Python em 24 Horas
Mais Informática	Banco de Dados: Aprenda de Forma Simples e F...
Mais Informática	Java para Que Te Quero
Companhia da Leitura	Minha Faculdade Vai Me Enlouquecer
Companhia da Leitura	Controlando as Emoções
Companhia da Leitura	Brasil Brasileiro
Ciência da Informação	Big Data: Conhecimentos Essenciais
Ciência da Informação	Ciência de Dados: O Futuro
Ciência da Informação	Inteligência Artificial Aplicada a Dados
Arco da Velha	Se Eu Voltasse no Passado
Arco da Velha	Penso, Logo Existo
Arco da Velha	Lugares para Viajar Sozinho

The Action Output pane shows the execution of the query, indicating that 15 rows were returned.

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, AVG(livro.preco)
FROM editora
JOIN livro ON editora.ideditora = livro.ideditora
GROUP BY editora.nome;
```

The screenshot shows the MySQL Workbench interface. The SQL Editor contains the following query:

```
1 SELECT editora.nome, AVG(livro.preco)
2 FROM editora
3 JOIN livro ON editora.ideditora = livro.ideditora
4 GROUP BY editora.nome;
```

The Result Grid displays the following data:

nome	AVG(livro.preco)
Companhia da Leitura	38.566667
Arco da Velha	56.233333
Mais Informática	106.566667
Sexta dos Estudos	76.566667
Ciência da Informação	147.233333

The Output tab shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
146	19:33:45	SELECT editoras.nome, AVG(livros.preco) AS me...	Error Code: 1146. Table 'livraria.editoras' doesn't ...	0.000 sec
147	19:34:24	SELECT editor.nome, AVG(livro.preco) FROM edi...	Error Code: 1054. Unknown column 'editor.nome' ...	0.000 sec
148	19:34:28	SELECT editora.nome, AVG(livro.preco) FROM e...	Error Code: 1054. Unknown column 'editora.id' in '...	0.000 sec
149	19:34:42	SELECT editora.nome, AVG(livro.preco) FROM e...	Error Code: 1054. Unknown column 'editora.idedit...	0.000 sec
150	19:35:03	SELECT editora.nome, AVG(livro.preco) FROM e...	5 row(s) returned	0.016 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, SUM(itemPedido.quantidade)
FROM Cliente
JOIN pedido ON Cliente.idCliente = pedido.idCliente
JOIN itemPedido ON pedido.idPedido = itemPedido.idPedido
GROUP BY Cliente.nome;
```

MySQL Workbench

SGBD Jeff - Warning - not su... x

File Edit View Query Database Server Tools Scripting Help

Navigator: SQL File 3* SQL File 3* SQL File 4* SQL File 5* SQL File 6* x

SCHEMAS

Filter objects

livraria

- Tables
- Views
- Stored Procedures
- Functions

sys

Administration Schemas

Information: No object selected

```

1 • SELECT Cliente.nome, SUM(itemPedido.quantidade)
2 FROM Cliente
3 JOIN pedido ON Cliente.idCliente = pedido.idCliente
4 JOIN itemPedido ON pedido.idPedido = itemPedido.idPedido
5 GROUP BY Cliente.nome;
6

```

Result Grid

nome	SUM(itemPedido.quantidade)
Alice de Souza	1
Mario Vicente	1
Maria Helena Mantovani	4
Vitor Martins	1
Nicole Amanda de Jesus	1
Luciano Tucolo	4
Paula Roberta Vitorino	2
Guilherme Koeriche	1
Beatriz Leopoldina	2
Lucas Cochuelo	2

Result 2 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
147	19:34:24	SELECT editora.nome, AVG(livro.preco) FROM edi...	Error Code: 1054. Unknown column 'editora.nome' ...	0.000 sec
148	19:34:28	SELECT editora.nome, AVG(livro.preco) FROM e...	Error Code: 1054. Unknown column 'editora.id' in '...	0.000 sec
149	19:34:42	SELECT editora.nome, AVG(livro.preco) FROM e...	Error Code: 1054. Unknown column 'editora.idedit...	0.000 sec
150	19:35:03	SELECT editora.nome, AVG(livro.preco) FROM e...	5 row(s) returned	0.016 sec / 0.000 sec
151	19:43:10	SELECT Cliente.nome, SUM(itemPedido.quantida...	10 row(s) returned	0.000 sec / 0.000 sec

Object Info Session