

Base de Dados

Base de Dados para OsLeitores



Pedro Emílio nº 52649
Luís Carvalho nº 51817
Gonçalo Prazeres nº52049

Departamento de Informática
Base de Dados
Docente: Irene Rodrigues

Conteúdo

1	Introdução	2
2	Exercícios e Resoluções	3
2.1	1 - Indicar as chaves primárias, candidatas e estrangeiras de cada relação.	3
2.2	2 - Indique os comandos SQL para a criação das tabelas que constituem esta base de dados. E construa esta base de dados no PostGres.	4
2.3	3 - Indique as expressões em SQL para inserir a seguinte informação na sua base de dados	
	a) → 20 membros (coloque valores adequados nos atributos)	5
2.4	b) → 10 livros e 5 autores. Cada autor deve ter pelo menos 2 livros. Um dos livros deve ter pelo menos 3 autores e outro 2 autores. Todos os livros devem ter o género policial e pelo menos mais um (drama, romance, ...).	5
2.5	c) → 1 membro deve ser amigo de todos os outros membros. 5 membros devem ter pelo menos 3 amigos.	6
2.6	d) → todos os membro devem gostar de pelo menos 1 livro. Pelo menos 1 membro deve gostar de todos os livros de um autor. E todos os membros devem ter lido pelo menos 3 livros.	7
2.7	e) → Em todas as perguntas abaixo, o resultado deve ter no mínimo um tuplo. Para isso deverá inserir mais valores nas suas tabelas de forma a garantir que a resposta não é vazia.	10
2.8	4 - Indique a expressão em SQL e em Algebra Relacional para responder às seguintes perguntas:	
	a) → Qual é o nome dos autores de obras do género romance	11
2.9	b) → Quais os membros que leram um livro do José Saramago?	11
2.10	c) → Qual o nome dos membros que leram e gostaram de livros do género romance?	11
2.11	d) → Qual o nome dos membros que leram e não gostaram de um livro?	11
2.12	e) → Qual o nome dos amigos do membro com o idMemb oleitor que gostaram de um livro de José Saramago?	12
2.13	f) → Quais os membros que não são amigos do membro com o idMemb oleitor?	12
2.14	g) → Qual o nome dos membros que gostam de livros do Lobo Antunes e do Francisco José Viegas? (ajuste os dados na sua base de dados para que exista pelo menos um membro a gostar dos dois livros)	12
2.15	h) → Quantos amigos tem o membro oleitor?	13
2.16	i) → Qual é o membro que tem mais amigos?	13
2.17	j) → Qual é o membro que gosta de mais livros?	13
2.18	k) → Qual o nome dos membros que são amigos do membro que gosta de mais livros	13
2.19	l) → Para cada livro indique o número de géneros.	13
2.20	m) → Para cada livro indique o número de géneros e o número de gostos	14
2.21	n) → Para cada autor indique o número de livros, o número de géneros e o número de gostos.	14
2.22	o) → Para cada membro, nome, indique o número de amigos e o número de livros de que gosta.	14
2.23	p) → Qual o nome dos membros que são amigos de todos os membros?	15
2.24	q) → Quais os títulos dos livros de que todos os amigos do leitor gostam?	15

Capítulo 1

Introdução

Pretende-se desenvolver uma base de dados para gerir a informação de uma rede social Os leitores.

Capítulo 2

Exercícios e Resoluções

2.1 1 - Indicar as chaves primárias, candidatas e estrangeiras de cada relação.

→Relação membro:

Chaves candidatas: {IdMemb} {IdMemb, Nome}

Chaves primárias: {IdMemb}

Chaves estrangeiras: {}

→Relação amigo:

Chaves candidatas: {IdMemb1, IdMemb2}

Chaves primárias: {IdMemb1, IdMemb2}

Chave estrangeiras: {IdMemb1, IdMemb2} da biblioteca membros

→Relação gosta:

Chaves candidatas: {IdMemb, ISBN}

Chaves primárias: {IdMemb, ISBN}

Chaves estrangeiras: {IdMemb} da biblioteca membros, {ISBN} da biblioteca livro

→Relação leu:

Chaves candidatas: {IdMemb, ISBN}

Chaves primárias: {IdMemb, ISBN}

Chaves estrangeiras: {IdMemb} da biblioteca membros, {ISBN} da biblioteca livro

→Relação livro:

Chaves candidatas: {ISBN}

Chaves primárias: {ISBN}

Chaves estrangeiras: {}

→Relação genero:

Chaves candidatas: {ISBN, Genero}

Chaves primárias: {ISBN, Genero}

Chaves estrangeiras: {ISBN} da biblioteca livro

→Relação autoria:

Chaves candidatas: {ISBN, Coda }

Chaves primárias: {ISBN, Coda}

Chaves estrangeiras: {ISBN} da biblioteca livro
{Coda} da biblioteca livro

→Relação autor:

Chaves candidatas: {Coda} {Coda, Nome}

Chaves primárias: {Coda}

Chaves estrangeiras: {}

2.2 2 - Indique os comandos SQL para a criação das tabelas que constituem esta base de dados. E construa esta base de dados no PostGres.

```
Create table membro ( Nome varchar(50), IdMemb char(15) primary key, Pais char(50) );
```

```
    Create table autor (  
Coda char(20) primary key,  
Nome varchar(50),  
Pais char(50)  
);
```

```
    Create table livro (  
ISBN varchar(50) primary key,  
Titulo char(50)  
);
```

```
    Create table amigo (  
IdMemb1 varchar(50),  
IdMemb2 varchar(50),  
primary key (IdMemb1, IdMemb2),  
foreign key (IdMemb1) references membro on delete restrict,  
foreign key (IdMemb2) references membro on delete restrict  
);
```

```
    Create table gosta (  
IdMemb varchar(50),  
ISBN char(50),  
primary key (IdMemb, ISBN),  
foreign key (IdMemb) references membro on delete restrict,  
foreign key (ISBN) references livro on delete restrict  
);
```

```
    Create table leu (  
IdMemb char(50),  
ISBN char(50),  
primary key (IdMemb, ISBN),  
foreign key (IdMemb) references membro on delete restrict,  
foreign key (ISBN) references livro on delete restrict  
);
```

```
    Create table genero (  
ISBN varchar(50),  
Genero char(20),  
primary key (ISBN, Genero),  
foreign key (ISBN) references livro on delete restrict  
);
```

```
    Create table autoria (  
ISBN varchar(50),  
Coda char(20),  
primary key (ISBN, Coda),  
foreign key (ISBN) references livro on delete restrict,  
foreign key (Coda) references autor on delete restrict  
);
```

2.3 3 - Indique as expressões em SQL para inserir a seguinte informação na sua base de dados

a) → 20 membros (coloque valores adequados nos atributos)

```
insert into membro values('Joaquim Alberto', 'af25ji67mn32500', 'Portugal');
insert into membro values('António Maria Juventude', 'af25ji67mn32501', 'Portugal');
insert into membro values('Maria das Couves', 'af25ji67mn32502', 'Portugal');
insert into membro values('Moreira Álvares', 'af25ji67mn32503', 'Portugal');
insert into membro values('Joana Barros', 'af25ji67mn32504', 'Portugal');
insert into membro values('Madalena Soares', 'af25ji67mn32505', 'Portugal');
insert into membro values('Luis Miguel', 'af25ji67mn32506', 'Portugal');
insert into membro values('Pedro dos Santos', 'af25ji67mn32507', 'Portugal');
insert into membro values('Joao Silva', 'af25ji67mn32508', 'Portugal');
insert into membro values('Carlos da Gama', 'af25ji67mn32509', 'Portugal');
insert into membro values('Ana Garcia', 'af25ji67mn32510', 'Portugal');
insert into membro values('Ines Sousa', 'af25ji67mn32511', 'Portugal');
insert into membro values('Barbara Silva', 'af25ji67mn32512', 'Portugal');
insert into membro values('Lourenço Nunes', 'af25ji67mn32513', 'Portugal');
insert into membro values('Teresa Castro', 'af25ji67mn32514', 'Portugal');
insert into membro values('Mariana Carvalho', 'af25ji67mn32515', 'Portugal');
insert into membro values('Tomás Rebelo', 'af25ji67mn32516', 'Portugal');
insert into membro values('Miguel Araujo', 'af25ji67mn32517', 'Portugal');
insert into membro values('Rui Barbosa', 'af25ji67mn32518', 'Portugal');
insert into membro values('Nicole', 'af25ji67mn32519', 'Portugal');
```

2.4 b) → 10 livros e 5 autores. Cada autor deve ter pelo menos 2 livros. Um dos livros deve ter pelo menos 3 autores e outro 2 autores. Todos os livros devem ter o género policial e pelo menos mais um (drama, romance, ...).

```
insert into livro values('978-3-16-148410-0', 'Ensaio sobre a Cegueira'); → José Saramago, Romance.
insert into livro values('978-3-16-148410-1', 'O mar em Casablanca'); → Francisco José Viegas.
insert into livro values('978-3-16-148410-2', 'Adeus, Minha Adorada'); → Policial e Romance - Raymond Chandle.
insert into livro values('978-3-16-148410-3', 'O Rapaz do Caixote de Madeira'); → Livro com 3 autores → Leon
Leyson e Marilyn J. Harran e Elizabeth B. Leyson.
insert into livro values('978-3-16-148410-4', 'A luz de Pequim'); → Francisco José Viegas.
insert into livro values('978-3-16-148410-5', 'Bom dia, Verônica'); → Livro com 2 autores → Ilana Casoy e Raphael
Montes.
insert into livro values('978-3-16-148410-6', 'Memorial do Convento'); → José Saramago, Romance.
insert into livro values('978-3-16-148410-7', 'As Crónicas'); → Lobo Antunes.
insert into livro values('978-3-16-148410-8', 'O Sono Eterno'); → Policial e Romance - Raymond Chandle, Romance.
insert into livro values('978-3-16-148410-9', 'As Naus: romance'); → Lobo Antunes, Romance.
insert into livro values('978-3-16-148411-0', 'O mar em Casablanca'); → Francisco José Viegas.
```

São 5 autores mas tivemos que adicionar mais dados como o pedido na linha e) para responder às questões, como é pedido um livro com 3 autores e outro com 2 autores

```
insert into autor values('0000-0000-0000-0000', 'Raymond Chandle', 'USA');
insert into autor values('0000-0000-0000-0001', 'Jose Saramago', 'Portugal');
insert into autor values('0000-0000-0000-0002', 'Francisco José Viegas', 'Portugal');
insert into autor values('0000-0000-0000-0003', 'Raphael Montes', 'Brasil');
insert into autor values('0000-0000-0000-0004', 'Lobo Antunes', 'Portugal');
insert into autor values('0000-0000-0000-0005', 'Ilana Casoy', 'Brasil');
insert into autor values('0000-0000-0000-0006', 'Leon Leyson', 'Polónia');
insert into autor values('0000-0000-0000-0007', 'Marilyn J. Harran', 'USA');
insert into autor values('0000-0000-0000-0008', 'Elizabeth B. Leyson', 'USA');
```

```
insert into autoria values('978-3-16-148410-0', '0000-0000-0000-0001');
```

```

insert into autoria values('978-3-16-148410-1' , '0000-0000-0000-0002');

insert into autoria values('978-3-16-148410-2' , '0000-0000-0000-0000');

insert into autoria values('978-3-16-148410-3' , '0000-0000-0000-0006');
insert into autoria values('978-3-16-148410-3' , '0000-0000-0000-0007');
insert into autoria values('978-3-16-148410-3' , '0000-0000-0000-0008');

insert into autoria values('978-3-16-148410-4' , '0000-0000-0000-0002');

insert into autoria values('978-3-16-148410-5' , '0000-0000-0000-0003');
insert into autoria values('978-3-16-148410-5' , '0000-0000-0000-0005');

insert into autoria values('978-3-16-148410-6' , '0000-0000-0000-0001');

insert into autoria values('978-3-16-148410-7' , '0000-0000-0000-0004');

insert into autoria values('978-3-16-148410-8' , '0000-0000-0000-0000');

insert into autoria values('978-3-16-148410-9' , '0000-0000-0000-0004');

insert into autoria values('978-3-16-148411-0' , '0000-0000-0000-0002');

insert into genero values('978-3-16-148410-0' , 'policial');
insert into genero values('978-3-16-148410-0' , 'romance');

insert into genero values('978-3-16-148410-1' , 'policial');

insert into genero values('978-3-16-148410-2' , 'policial');
insert into genero values('978-3-16-148410-2' , 'romance');

insert into genero values('978-3-16-148410-3' , 'policial');
insert into genero values('978-3-16-148410-3' , 'Biografia');

insert into genero values('978-3-16-148410-4' , 'policial');
insert into genero values('978-3-16-148410-4' , 'mistério');

insert into genero values('978-3-16-148410-5' , 'policial');
insert into genero values('978-3-16-148410-5' , 'mistério');
insert into genero values('978-3-16-148410-5' , 'drama');

insert into genero values('978-3-16-148410-6' , 'policial');
insert into genero values('978-3-16-148410-6' , 'romance');

insert into genero values('978-3-16-148410-7' , 'policial');

insert into genero values('978-3-16-148410-8' , 'policial');
insert into genero values('978-3-16-148410-8' , 'romance');
insert into genero values('978-3-16-148410-8' , 'mistério');

insert into genero values('978-3-16-148410-9' , 'policial'); insert into genero values('978-3-16-148410-9' , 'romance');
insert into genero values('978-3-16-148411-0' , 'policial'); insert into genero values('978-3-16-148411-0' , 'mistério');

```

2.5 c) → 1 membro deve ser amigo de todos os outros membros. 5 membros devem ter pelo menos 3 amigos.

```

1 membro é amigo de todos
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32500');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32501');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32502');

```

```

insert into amigo values('af25ji67mn32519' , 'af25ji67mn32503');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32504');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32505');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32506');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32507');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32508');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32509');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32510');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32511');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32512');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32513');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32514');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32515');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32516');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32517');
insert into amigo values('af25ji67mn32519' , 'af25ji67mn32518');

```

5 membros têm 3 amigos

```

insert into amigo values('af25ji67mn32514' , 'af25ji67mn32500');
insert into amigo values('af25ji67mn32514' , 'af25ji67mn32501');
insert into amigo values('af25ji67mn32514' , 'af25ji67mn32502');

```

```

insert into amigo values('af25ji67mn32516' , 'af25ji67mn32503');
insert into amigo values('af25ji67mn32516' , 'af25ji67mn32504');
insert into amigo values('af25ji67mn32516' , 'af25ji67mn32506');

```

```

insert into amigo values('af25ji67mn32511' , 'af25ji67mn32503');
insert into amigo values('af25ji67mn32511' , 'af25ji67mn32510');
insert into amigo values('af25ji67mn32511' , 'af25ji67mn32509');

```

```

insert into amigo values('af25ji67mn32503' , 'af25ji67mn32501');
insert into amigo values('af25ji67mn32503' , 'af25ji67mn32504');
insert into amigo values('af25ji67mn32503' , 'af25ji67mn32508');

```

```

insert into amigo values('af25ji67mn32501' , 'af25ji67mn32505');
insert into amigo values('af25ji67mn32501' , 'af25ji67mn32512');
insert into amigo values('af25ji67mn32501' , 'af25ji67mn32508');

```

2.6 d) → todos os membro devem gostar de pelo menos 1 livro. Pelo menos 1 membro deve gostar de todos os livros de um autor. E todos os membros devem ter lido pelo menos 3 livros.

Todos os membros gostam de pelo menos um livro NOTA: Faz sentido ter lido para poder gostar.

```

insert into gosta values('af25ji67mn32500' , '978-3-16-148410-3' );
insert into gosta values('af25ji67mn32501' , '978-3-16-148410-9');
insert into gosta values('af25ji67mn32502' , '978-3-16-148410-2');
insert into gosta values('af25ji67mn32503' , '978-3-16-148410-7');
insert into gosta values('af25ji67mn32504' , '978-3-16-148410-8');
insert into gosta values('af25ji67mn32505' , '978-3-16-148410-4');
insert into gosta values('af25ji67mn32506' , '978-3-16-148410-9');
insert into gosta values('af25ji67mn32507' , '978-3-16-148410-5');
insert into gosta values('af25ji67mn32508' , '978-3-16-148410-1');
insert into gosta values('af25ji67mn32509' , '978-3-16-148410-0');
insert into gosta values('af25ji67mn32510' , '978-3-16-148410-2');
insert into gosta values('af25ji67mn32511' , '978-3-16-148410-5');
insert into gosta values('af25ji67mn32512' , '978-3-16-148410-8');
insert into gosta values('af25ji67mn32513' , '978-3-16-148410-6');
insert into gosta values('af25ji67mn32514' , '978-3-16-148410-3');
insert into gosta values('af25ji67mn32515' , '978-3-16-148410-0');
insert into gosta values('af25ji67mn32516' , '978-3-16-148410-8');

```



```

insert into gosta values('af25ji67mn32517' , '978-3-16-148410-7');
insert into gosta values('af25ji67mn32518' , '978-3-16-148410-9');
insert into gosta values('af25ji67mn32519' , '978-3-16-148410-4');
insert into gosta values('af25ji67mn32502' , '978-3-16-148410-9');
insert into gosta values('af25ji67mn32502' , '978-3-16-148410-4');
insert into gosta values('af25ji67mn32502' , '978-3-16-148410-3');
insert into gosta values('af25ji67mn32502' , '978-3-16-148410-7');
insert into gosta values('af25ji67mn32518' , '978-3-16-148410-6');
insert into gosta values('af25ji67mn32500' , '978-3-16-148410-0' );

```

Pelo menos um membro deve gostar de todos os livros de um autor

José Saramago

```

insert into gosta values('af25ji67mn32504' , '978-3-16-148410-0');
insert into gosta values('af25ji67mn32504' , '978-3-16-148410-6');
insert into gosta values('af25ji67mn32508' , '978-3-16-148410-0');
insert into gosta values('af25ji67mn32508' , '978-3-16-148410-6');
Francisco José Viegas

```

```

insert into gosta values('af25ji67mn32513' , '978-3-16-148410-1');
insert into gosta values('af25ji67mn32513' , '978-3-16-148410-4');
insert into gosta values('af25ji67mn32513' , '978-3-16-148411-0');
insert into gosta values('af25ji67mn32518' , '978-3-16-148410-1');
insert into gosta values('af25ji67mn32518' , '978-3-16-148410-4');
insert into gosta values('af25ji67mn32518' , '978-3-16-148411-0');
insert into gosta values('af25ji67mn32500' , '978-3-16-148410-1');
insert into gosta values('af25ji67mn32500' , '978-3-16-148410-4');
insert into gosta values('af25ji67mn32500' , '978-3-16-148411-0');

```

Todos os membros devem ter lido pelo menos 3 livros

```

insert into leu values('af25ji67mn32500' , '978-3-16-148410-7' );
insert into leu values('af25ji67mn32500' , '978-3-16-148410-1' );
insert into leu values('af25ji67mn32500' , '978-3-16-148410-4' );
insert into leu values('af25ji67mn32500' , '978-3-16-148410-5' );
insert into leu values('af25ji67mn32500' , '978-3-16-148410-0' );
insert into leu values('af25ji67mn32500' , '978-3-16-148411-0' );
insert into leu values('af25ji67mn32500' , '978-3-16-148410-3' );

```

```

insert into leu values('af25ji67mn32501' , '978-3-16-148410-2' );
insert into leu values('af25ji67mn32501' , '978-3-16-148410-5' );
insert into leu values('af25ji67mn32501' , '978-3-16-148410-4' );
insert into leu values('af25ji67mn32501' , '978-3-16-148410-8' );
insert into leu values('af25ji67mn32501' , '978-3-16-148410-9' );

```

```

insert into leu values('af25ji67mn32502' , '978-3-16-148410-9' );
insert into leu values('af25ji67mn32502' , '978-3-16-148410-4' );
insert into leu values('af25ji67mn32502' , '978-3-16-148410-3' );
insert into leu values('af25ji67mn32502' , '978-3-16-148410-7' );
insert into leu values('af25ji67mn32502' , '978-3-16-148410-2' );

```

```

insert into leu values('af25ji67mn32503' , '978-3-16-148410-7' );
insert into leu values('af25ji67mn32503' , '978-3-16-148410-2' );
insert into leu values('af25ji67mn32503' , '978-3-16-148410-6' );
insert into leu values('af25ji67mn32503' , '978-3-16-148411-0' );
insert into leu values('af25ji67mn32503' , '978-3-16-148410-0' );

```

```

insert into leu values('af25ji67mn32504' , '978-3-16-148410-2' );
insert into leu values('af25ji67mn32504' , '978-3-16-148410-1' );
insert into leu values('af25ji67mn32504' , '978-3-16-148410-4' );
insert into leu values('af25ji67mn32504' , '978-3-16-148410-5' );
insert into leu values('af25ji67mn32504' , '978-3-16-148410-7' );
insert into leu values('af25ji67mn32504' , '978-3-16-148410-0' );
insert into leu values('af25ji67mn32504' , '978-3-16-148410-6' );
insert into leu values('af25ji67mn32504' , '978-3-16-148410-9' );

```



```

insert into leu values('af25ji67mn32514' , '978-3-16-148410-9' );

insert into leu values('af25ji67mn32515' , '978-3-16-148410-7' );
insert into leu values('af25ji67mn32515' , '978-3-16-148410-9' );
insert into leu values('af25ji67mn32515' , '978-3-16-148410-6' );
insert into leu values('af25ji67mn32515' , '978-3-16-148410-0' );
insert into leu values('af25ji67mn32515' , '978-3-16-148410-1' );
insert into leu values('af25ji67mn32515' , '978-3-16-148410-4' );

insert into leu values('af25ji67mn32516' , '978-3-16-148410-5' );
insert into leu values('af25ji67mn32516' , '978-3-16-148410-3' );
insert into leu values('af25ji67mn32516' , '978-3-16-148410-2' );
insert into leu values('af25ji67mn32516' , '978-3-16-148410-0' );
insert into leu values('af25ji67mn32516' , '978-3-16-148410-8' );

insert into leu values('af25ji67mn32517' , '978-3-16-148410-7' );
insert into leu values('af25ji67mn32517' , '978-3-16-148410-4' );
insert into leu values('af25ji67mn32517' , '978-3-16-148410-2' );
insert into leu values('af25ji67mn32517' , '978-3-16-148411-0' );

insert into leu values('af25ji67mn32518' , '978-3-16-148410-8' );
insert into leu values('af25ji67mn32518' , '978-3-16-148410-2' );
insert into leu values('af25ji67mn32518' , '978-3-16-148410-6' );
insert into leu values('af25ji67mn32518' , '978-3-16-148410-4' );
insert into leu values('af25ji67mn32518' , '978-3-16-148410-3' );
insert into leu values('af25ji67mn32518' , '978-3-16-148410-5' );
insert into leu values('af25ji67mn32518' , '978-3-16-148411-0' );
insert into leu values('af25ji67mn32518' , '978-3-16-148410-9' );

insert into leu values('af25ji67mn32519' , '978-3-16-148410-6' );
insert into leu values('af25ji67mn32519' , '978-3-16-148410-1' );
insert into leu values('af25ji67mn32519' , '978-3-16-148410-8' );
insert into leu values('af25ji67mn32519' , '978-3-16-148410-9' );
insert into leu values('af25ji67mn32519' , '978-3-16-148411-0' );
insert into leu values('af25ji67mn32519' , '978-3-16-148410-4' );

```

2.7 e) → Em todas as perguntas abaixo, o resultado deve ter no mínimo um tuplo. Para isso deverá inserir mais valores nas suas tabelas de forma a garantir que a resposta não é vazia.

Foi adicionado o membro com idmemb "oleitor" para responder às alíneas da questão 4

```

insert into membro values('Sr.Oleitor' , 'oleitor' , 'Portugal');

```

```

insert into amigo values('oleitor' , 'af25ji67mn32500');
insert into amigo values('oleitor' , 'af25ji67mn32501');
insert into amigo values('oleitor' , 'af25ji67mn32506');
insert into amigo values('oleitor' , 'af25ji67mn32508');
insert into amigo values('oleitor' , 'af25ji67mn32512');
insert into amigo values('oleitor' , 'af25ji67mn32513');
insert into amigo values('oleitor' , 'af25ji67mn32514');
insert into amigo values('oleitor' , 'af25ji67mn32515');
insert into amigo values('oleitor' , 'af25ji67mn32518');
insert into amigo values('oleitor' , 'af25ji67mn32519');

```

```

insert into gosta values('oleitor' , '978-3-16-148410-9');
insert into gosta values('oleitor' , '978-3-16-148411-0');
insert into gosta values('oleitor' , '978-3-16-148410-4');
insert into gosta values('oleitor' , '978-3-16-148410-8');
insert into gosta values('oleitor' , '978-3-16-148410-6');

```

```

insert into leu values('oleitor' , '978-3-16-148410-9');
insert into leu values('oleitor' , '978-3-16-148410-0');
insert into leu values('oleitor' , '978-3-16-148411-0');
insert into leu values('oleitor' , '978-3-16-148410-6');
insert into leu values('oleitor' , '978-3-16-148410-3');
insert into leu values('oleitor' , '978-3-16-148410-4');
insert into leu values('oleitor' , '978-3-16-148410-8');
insert into leu values('oleitor' , '978-3-16-148410-1');

```

2.8 4 - Indique a expressão em SQL e em Algebra Relacional para responder às seguintes perguntas:

a) → Qual é o nome dos autores de obras do género romance

SQL

```

SELECT DISTINCT nome
FROM autor NATURAL INNER JOIN autoria NATURAL INNER JOIN genero
WHERE genero = 'romance'

```

Algebra relacional

π nome (σ genero = "romance" (autor \bowtie autoria \bowtie genero))

2.9 b) → Quais os membros que leram um livro do José Saramago?

SQL

```

SELECT DISTINCT nome
FROM membro NATURAL INNER JOIN leu NATURAL INNER JOIN autoria
WHERE Coda = '0000-0000-0000-0001'

```

Algebra relacional

π nome (σ Coda = "0000-0000-0000-0001" (membro \bowtie leu \bowtie autoria))

2.10 c) → Qual o nome dos membros que leram e gostaram de livros do género romance?

SQL

```

SELECT DISTINCT nome
FROM membro NATURAL INNER JOIN leu NATURAL INNER JOIN genero
WHERE genero = 'romance'
INTERSECT
SELECT DISTINCT nome
FROM membro NATURAL INNER JOIN gosta NATURAL INNER JOIN genero
WHERE genero = 'romance'

```

Algebra relacional

π nome (σ genero = "romance" (membro \bowtie leu \bowtie gosta \bowtie genero))

2.11 d) → Qual o nome dos membros que leram e não gostaram de um livro?

SQL

```

SELECT nome
FROM membro NATURAL INNER JOIN leu
EXCEPT
SELECT nome

```

FROM membro NATURAL INNER JOIN leu NATURAL INNER JOIN gosta

Algebra relacional

$\pi \text{ nome}(\text{membro} \bowtie \text{leu}) - \pi \text{ nome}(\text{membro} \bowtie \text{leu} \bowtie \text{gosta})$

2.12 e) → Qual o nome dos amigos do membro com o idMemb oleitor que gostaram de um livro de José Saramago?

SQL

```
SELECT DISTINCT Nome → É amigo do oleitor
FROM membro NATURAL INNER JOIN amigo WHERE IdMemb1 = 'oleitor'
INTERSECT
SELECT DISTINCT Nome → Gostaram de um livro de José Saramago
FROM membro NATURAL INNER JOIN gosta NATURAL INNER JOIN autoria
WHERE Coda = '0000-0000-0000-0001'
```

Algebra relacional

$\pi \text{ nome} (\sigma \text{ IdMemb1} = \text{"oleitor"}(\text{membro} \bowtie \text{amigo})) \cap \pi \text{ nome} (\sigma \text{ Coda} = \text{"0000-0000-0000-0001"}(\text{membro} \bowtie \text{gosta} \bowtie \text{autoria}))$

2.13 f) → Quais os membros que não são amigos do membro com o idMemb oleitor?

SQL

```
SELECT DISTINCT nome, idmemb
FROM membro NATURAL INNER JOIN amigo
EXCEPT
SELECT DISTINCT nome, idmemb2
FROM membro NATURAL INNER JOIN amigo WHERE idmemb1 = 'oleitor'
```

Algebra relacional

$\pi \text{ nome, idmemb} (\text{membro} \bowtie \text{amigo}) - \pi \text{ nome, idmemb2} (\sigma \text{ IdMemb1} = \text{"oleitor"}(\text{membro} \bowtie \text{amigo}))$

2.14 g) → Qual o nome dos membros que gostam de livros do Lobo Antunes e do Francisco José Viegas? (ajuste os dados na sua base de dados para que exista pelo menos um membro a gostar dos dois livros)

SQL

```
SELECT DISTINCT nome
FROM membro NATURAL INNER JOIN gosta NATURAL INNER JOIN autoria
WHERE coda = '0000-0000-0000-0002' → Código Francisco José Viegas
INTERSECT
SELECT DISTINCT nome
FROM membro NATURAL INNER JOIN gosta NATURAL INNER JOIN autoria
WHERE coda = '0000-0000-0000-0004' → Código Lobo Antunes
```

Algebra relacional

$\pi \text{ nome} (\sigma \text{ coda} = \text{"0000-0000-0000-0002"}(\text{membro} \bowtie \text{gosta} \bowtie \text{autoria})) \cap \pi \text{ nome} (\sigma \text{ Coda} = \text{"0000-0000-0000-0004"}(\text{membro} \bowtie \text{gosta} \bowtie \text{autoria}))$

2.15 h) → Quantos amigos tem o membro oleitor?

SQL

```
SELECT COUNT(distinct idmemb2)
FROM amigo
WHERE idmemb1 = 'oleitor'
```

Algebra relacional

$R \leftarrow \text{amigo}$

$S \leftarrow \sigma_{\text{idmemb1} = \text{"oleitor"}} \text{amigo}(R)$

$G \text{ count}(\text{idmemb2}) \text{ as } n(S) \rightarrow \text{Agregação}$

2.16 i) → Qual é o membro que tem mais amigos?

SQL

```
SELECT MAX(idmemb2) → Idmemb do membro com mais amigos
FROM amigo
```

Algebra relacional

Não foi feita a Algebra Relacional.

2.17 j) → Qual é o membro que gosta de mais livros?

SQL

```
SELECT MAX(idmemb) → Idmemb do membro que gosta de mais livros
FROM membro NATURAL INNER JOIN gosta
```

Algebra relacional

Não foi feita a Algebra Relacional.

2.18 k) → Qual o nome dos membros que são amigos do membro que gosta de mais livros

SQL

```
WITH contarAmigos(idmemb, livro) as (select gosta.idmemb, COUNT(gosta.isbn) as livro
From gosta
GROUP BY idmemb)
```

```
SELECT DISTINCT idmemb2
from(select max(livro) as livro
from contarAmigos) as gosta, amigo, contarAmigos
where gosta.livro=contarAmigos.livro and amigo.idmemb1=contarAmigos.idmemb;
```

Algebra relacional

Não foi feita a Algebra Relacional.

2.19 l) → Para cada livro indique o número de géneros.

SQL

```
SELECT DISTINCT titulo, count (genero)
from livro, genero
where livro.isbn = genero.isbn group by titulo;
```

Algebra relacional

π titulo, COUNT (genero) (σ livro . isbn = genero . isbn (livro \times genero))

2.20 m) \rightarrow Para cada livro indique o número de géneros e o número de gostos

SQL

```
SELECT DISTINCT titulo,  
count(genero.isbn) as numeroDeGeneros, count(gosta.isbn) as numeroDeGostos
```

```
FROM livro NATURAL INNER join genero NATURAL INNER join gosta  
WHERE livro.isbn=gosta.isbn AND genero.isbn=gosta.isbn AND livro.isbn=genero.isbn  
GROUP BY titulo;
```

Algebra relacional

π titulo, COUNT (isbn) \rightarrow numerodegeneros, COUNT (isbn) \rightarrow numerodegostos

γ titulo, COUNT (isbn)

σ livro . isbn = gosta . isbn \wedge genero . isbn = gosta .

isbn \wedge livro . isbn = genero . isbn (livro \bowtie genero \bowtie gosta)

2.21 n) \rightarrow Para cada autor indique o número de livros, o número de géneros e o número de gostos.

SQL

```
SELECT DISTINCT autor.nome,  
count(livro.isbn) as numerosDeLivros,  
count(genero.isbn) as numeroDeGeneros,  
count(gosta.isbn) as numeroDeGostos
```

```
FROM autor NATURAL INNER join autoria NATURAL INNER join  
livro NATURAL INNER join gosta NATURAL INNER join genero  
WHERE autor.coda = autoria.coda AND autoria.isbn =  
gosta.isbn AND autoria.isbn = livro.isbn AND autoria.isbn  
= genero.isbn  
GROUP BY autor.nome;
```

Algebra relacional

π autor . nome, COUNT (isbn) \rightarrow numerosdelivros, COUNT (isbn) \rightarrow numerodegeneros, COUNT (isbn) \rightarrow numerodegostos

γ nome, COUNT (isbn)

σ autor . coda = autoria . coda \wedge autoria . isbn = gosta . isbn \wedge autoria . isbn = livro . isbn \wedge autoria . isbn = genero . isbn (autor \bowtie autoria \bowtie livro \bowtie gosta \bowtie genero)

2.22 o) \rightarrow Para cada membro, nome, indique o número de amigos e o número de livros de que gosta.

SQL.

```
SELECT DISTINCT nome,  
count(amigo.idmemb2) as numeroDeAmigos,  
count(gosta.isbn) as numeroDeLivrosQueGosta
```

```
FROM membro NATURAL INNER join amigo NATURAL INNER join gosta  
WHERE membro.idmemb=amigo.idmemb1 AND  
membro.idmemb=gosta.idmemb  
GROUP BY nome;
```

Algebra relacional

π nome, COUNT (idmemb2) \rightarrow numerodeamigos, COUNT (isbn) \rightarrow numerodelivrosquegosta
 γ nome, COUNT (idmemb2), COUNT (isbn)
 σ membro . idmemb = amigo . idmemb1 \wedge membro . idmemb = gosta . idmemb (membro \bowtie amigo \bowtie gosta)

2.23 p) \rightarrow Qual o nome dos membros que são amigos de todos os membros?

SQL
 SELECT DISTINCT nome
 FROM membro as amigo1
 where not exists (SELECT idmemb
 FROM membro
 WHERE membro.idmemb != amigo1.idmemb
 EXCEPT(SELECT idmemb1
 FROM amigo
 WHERE amigo.idmemb2 = amigo1.idmemb
 UNION
 SELECT idmemb2
 FROM amigo
 WHERE amigo.idmemb1 = amigo1.idmemb));

Algebra relacional
 Não foi feita a Algebra Relacional.

2.24 q) \rightarrow Quais os titulos dos livros de que todos os amigos do leitor gostam?

SQL
 WITH contarGostos (idmemb) as (Select gosta.idmemb FROM gosta)

 SELECT DISTINCT titulo
 FROM count_NATURAL INNER join livro NATURAL INNER join amigo
 WHERE amigo.idmemb1='oleitor' AND
 amigo.idmemb2=count_.idmemb

Algebra relacional
 Não foi feita a Algebra Relacional.