

U. C. Base de Dados

Licenciatura em Engenharia Informática

Docente:

Irene Rodrigues

Discentes:

André Baião, 48092

Gonçalo Barradas, 48402



Índice

Introdução	1
Exercícios	2
Exercício 1	2
Exercício 2	2
Exercício 3	3
Exercício 4	4
Exercício 5	4
Exercício 6	4
Exercício 7	5
Exercício 8	5
Exercício 9	5
Exercício 10	5
Exercício 10. a)	5
Exercício 10. b)	6
Exercício 10. c)	6
Exercício 10. d)	7
Exercício 10. e)	7
Exercício 10. f)	7
Exercício 10. g)	8
Exercício 10. h)	8
Exercício 10. i)	8
Exercício 10. j)	10
Exercício 10. k)	11
Exercício 10. l)	11
Exercício 10. m)	11
Exercício 10. n)	12



Exercício 10. o).....	13
Exercício 10. p).....	13

Introdução

Com o propósito de avaliar os conhecimentos adquiridos ao longo deste semestre referentes aos diagramas E-R (Entidade-Relação), dependências funcionais e a forma normal de Boyce Codd, e SQL, foi-nos proposto um trabalho que consiste na resolução de vários exercícios.

Um diagrama Entidade-Relação é um tipo de fluxograma que ilustra como “entidades” pessoas, objetos ou conceitos que se relacionam entre si dentro de um sistema.

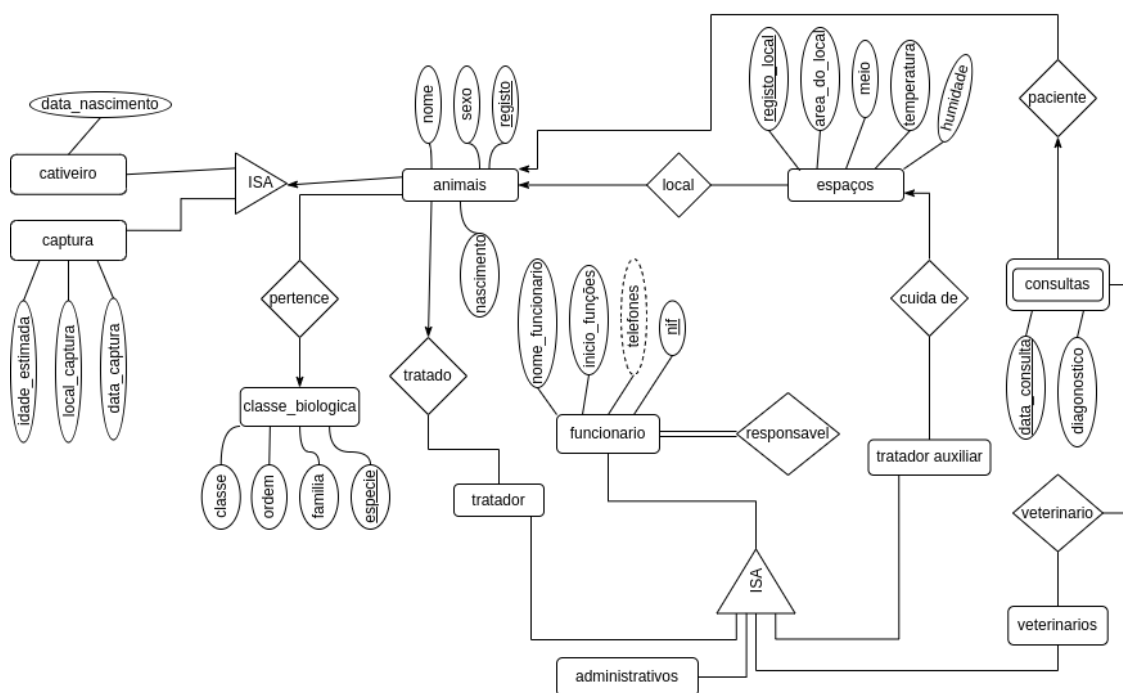
As dependências funcionais permitem restringir o número de relações válidas.



Exercícios

Exercício 1

Para o problema acima, construa um diagrama Entidade-Relação que descreva a informação. No diagrama não se esqueça de indicar as restrições das relações e as chaves primárias das entidades.



Exercício 2

Transforme o modelo E-R em tabelas. Nas tabelas não se esqueça de indicar as chaves primárias e as chaves candidatas.

- **animais** (registo, nome, sexo, local)
- **classe_bio** (especie, classe, ordem, familia, registo)
- **captura** (registo, local_captura, data_captura, idade_estimada)
- **cativoiro** (registo, registo_mae, registo_pai)
- **espacos** (registo_local, area, meio, clima)



Exercício 2 (continuação)

- **funcionario** (nif, nome_func, inicio_func)
- **telf_funcionario** (nif, telemovel)
- **responsavel** (nif_funcionario, nif_responsvel)
- **tratador** (registo, nif)
- **tratador_auxiliar** (registo_local, nif)
- **veterinarios** (nif)
- **consultas** (nif, data_consulta, registo, registo_local, diagnostico)

Exercício 3

Defina o conjunto de dependências funcionais que a base de dados deve verificar.

Exemplo, as seguintes dependências funcionais devem-se verificar neste domínio:

- local→tratador_auxiliar
- animal→tratador
- espécie→família
- família→ordem
- ordem→classe
- animal→registo
- animal→tratador
- espécie→família
- família→ordem
- ordem→classe
- local→tratador_auxiliar
- funcionario→responsavel
- funcionario→nif
- funcionario→telefonos
- consulta→veterinario
- consulta→animal
- consulta→local



Exercício 4

Apresente a Base de Dados na forma normal de Boyce Codd, BCNF.

registro \rightarrow nome, sexo, local;

especie \rightarrow classe, ordem, familia, registro;

registro \rightarrow especie;

registro \rightarrow local_captura, data_captura, idade_estimada;

registro \rightarrow registro_mae, registro_pai;

registro_local \rightarrow area, meio, clima;

nif \rightarrow nome_func, inicio_func;

nif_funcionario \rightarrow nif_responsvel;

registro \rightarrow nif;

registro_local \rightarrow nif;

Sendo $A \rightarrow B$ uma dependência funcional, podemos afirmar que a base de dados está na forma normal de Boyce-Codd visto que A é super-chave das relações demonstradas.

Exercício 5

Indique justificando se a sua base de dados tem informação redundante.

Sim, a base de dados acima apresenta informação redundante pois na tabela animais o argumento nascimento é desnecessário visto que conseguiríamos obter o seu conteúdo através de uma união com as tabelas cativo e captura.

Exercício 6

Indique se a base de dados que apresentou na forma normal de Boyce-Codd preserva as dependências.

A base de dados na forma normal de Boyce-Codd preserva as dependências.

Exercício 7

Para cada relação da base de dados na BCNF indique as chaves primárias, candidatas e estrangeiras.

Exercício 8

Indique os comandos SQL para a criação das tabelas que constituem esta base de dados. E construa esta base de dados no PostgreSQL.

Resposta no ficheiro em anexo.

Exercício 9

Indique as expressões em SQL para inserir a seguinte informação na sua base de dados e inseria-a.

Resposta no ficheiro em anexo.

Exercício 10

Indique a expressão em SQL para responder às seguintes perguntas (no relatório indique também o resultado).

Exercício 10. a)

Em que locais do zoo se podem visitar aves?

SELECT DISTINCT local FROM animais

NATURAL INNER JOIN classe_bio

WHERE classe LIKE 'aves'; Exercício 10;

registo_local
A6

Exercício 10. b)

Em que locais do zoo não há carnívoros?

```
SELECT DISTINCT local FROM animais
```

```
NATURAL INNER JOIN classe_bio
```

```
WHERE ordem NOT LIKE 'carnívoros';
```

registo_local
A6
A2
A1

Exercício 10. c)

Indique os irmãos da Kilu.

```
SELECT nome FROM animais
```

```
NATURAL INNER JOIN cativoiro
```

```
WHERE registo_pai = (
```

```
  SELECT registo_pai FROM cativoiro
```

```
  NATURAL INNER JOIN animais
```

```
  WHERE animais.nome = 'Kilu')
```

```
OR
```

```
registo_mae = (SELECT registo_mae FROM cativoiro
```

```
  NATURAL INNER JOIN animais
```

```
  WHERE animais.nome = 'Kilu')
```

```
EXCEPT
```

```
SELECT nome FROM animais
```

```
WHERE nome = 'Kilu';
```

nome
Kulu

Exercício 10. d)

Indique os telefones do tratador responsável pela Kata.

```
SELECT numero FROM telefones  
NATURAL INNER JOIN tratador  
NATURAL INNER JOIN animais  
WHERE animais.nome LIKE 'Kata';
```

numero
266787809
919999999

Exercício 10. e)

Indique os telefones do responsável pelo auxiliar responsável pela local onde está a Kata.

```
WITH aux AS (SELECT nif FROM tratador_auxiliar  
NATURAL INNER JOIN animais  
WHERE tratador_auxiliar.registo_local = animais.local AND nome = 'Kata')  
SELECT numero FROM telefones  
NATURAL INNER JOIN aux  
NATURAL INNER JOIN responsavel  
WHERE responsavel.nif_funcionario = aux.nif;
```

numero
919999996
266878806

Exercício 10. f)

Indique os tratamentos (data e tratamento) que a Mali já fez no zoo.

```
SELECT data_consulta, diagnostico FROM consultas  
NATURAL INNER JOIN animais  
WHERE nome LIKE 'Mali'
```

data_consulta	diagnostico
2005-08-12	grávida
2005-09-12	cálcio injectado
2005-12-12	parto
2006-07-12	infecção
2006-07-12	antibiótico injectado

Exercício 10. g)

Indique os nomes dos veterinários que já diagnosticaram uma gravidez a um carnívoro.

```
SELECT nome_func FROM funcionario
JOIN consultas ON consultas.nif = funcionario.nif
JOIN classe_bio ON consultas.registo = classe_bio.registo
WHERE ordem LIKE 'carnívoros'
AND
diagnostico LIKE 'grávida';
```

nome_funcionario
Pedro Vale

Exercício 10. h)

Indique para cada família da ordem artiodáctilos quantos animais tem o zoo.

```
SELECT familia, COUNT(ordem) AS numAnimais
FROM classe_bio
WHERE ordem LIKE 'artiodáctilos'
GROUP BY familia;
```

familia	numAnimais
cervídeos	5
hipopótamos	3

Exercício 10. i)

Indique para cada espécie quais os pares de animais que podem ser acasalados, sabendo que não se devem acasalar pais com filhos ou irmãos.

```
WITH fem(femName,registo,especie) AS (SELECT animais.nome,animais.registo, especie
FROM animais
NATURAL INNER JOIN classe_bio
WHERE sexo LIKE 'feminino'
ORDER BY especie),
mas(masName, registo,especie) AS (SELECT animais.nome, registo, especie FROM animais
NATURAL INNER JOIN classe_bio
WHERE sexo LIKE 'masculino'
ORDER BY especie),
```



```
catfem(nome, registo, registo_mae, registo_pai, especie) AS
(SELECT animais.nome , animais.registo, registo_mae, registo_pai, classe_bio.especie
FROM catveiro NATURAL INNER JOIN animais
NATURAL INNER JOIN classe_bio
Where sexo like 'feminino'),
catmas(nome, registo, registo_mae, registo_pai, especie) AS
(SELECT animais.nome , animais.registo, registo_mae, registo_pai, classe_bio.especie
FROM catveiro NATURAL INNER JOIN animais
NATURAL INNER JOIN classe_bio
Where sexo like 'masculino')
SELECT femName , masname, fem.especie FROM fem
JOIN mas ON fem.especie=mas.especie
EXCEPT
SELECT fem.femName , animais.nome, fem.especie FROM fem
JOIN catveiro ON fem.registo = catveiro.registo
JOIN animais ON catveiro.registo_pai= animais.registo
EXCEPT
SELECT animais.nome, mas.masName , mas.especie FROM mas
JOIN catveiro ON mas.registo = catveiro.registo
JOIN animais ON catveiro.registo_mae= animais.registo
EXCEPT
SELECT animais.nome, catmas.nome , catmas.especie FROM catmas
JOIN catveiro on catmas.registo_mae=catveiro.registo_mae
JOIN animais on catveiro.registo=animais.registo
EXCEPT
SELECT animais.nome, catmas.nome ,catmas.especie FROM catmas
JOIN catveiro on catmas.registo_pai=catveiro.registo_pai
JOIN animais on catveiro.registo=animais.registo
```

Feminino	Masculino	especie
Zula	Zará	arara-azul-pequena
Rará	Zará	arara-azul-pequena
Zura	Zará	arara-azul-pequena
Rará	Ará	arara-azul-pequena
Tapi	Hipo	hipopótamo comum
Luka	Kaki	veado
Kalu	Kaki	veado
Kalu	Kuli	veado
Mali	Mata	tigre
Mali	Cáta	tigre
Aka	Cáta	tigre
Kata	Cáta	tigre
Kata	Taji	tigre
Mali	Taji	tigre

Exercício 10. j)

SELECT ordem, COUNT(ordem) AS num

FROM class_bio

GROUP BY ordem

ORDER BY num DESC

LIMIT 1;

ordem	num
artiodáctilos	8

Exercício 10. k)

Qual é a ordem dos animais que têm mais de 5 consultas por ano (diagnóstico ou tratamento).

```
SELECT ordem, COUNT(ordem) FROM classe_bio
```

```
NATURAL INNER JOIN consultas
```

```
WHERE classe_bio.registo = consultas.registo
```

```
GROUP BY ordem
```

```
HAVING COUNT(ordem)>5;
```

ordem	count
artiodáctilos	12
psittaciformes	6

Exercício 10. l)

Indique o número de animais nascidos em cativeiro.

```
WITH anms AS (SELECT registo, COUNT(registo) AS cat FROM cativeiro
```

```
GROUP BY registo)
```

```
SELECT SUM(cat) FROM anms;
```

SUM
10

Exercício 10. m)

Qual é o animal (nome e espécie) mais velho do zoo?

```
WITH a AS (SELECT data_nascimento AS dn , registo
```

```
FROM cativeiro
```

```
UNION
```

```
SELECT idd_estimada AS dn , registo
```

```
FROM captura)
```

```
SELECT nome FROM animais
```

```
WHERE registo=
```

```
(SELECT registo
```

```
FROM a
```

```
WHERE dn=(SELECT MIN(dn)
```

```
FROM a));
```

maisVelho

Hipo

Exercício 10. n)

Qual é o local húmido com mais mamíferos?

```
WITH mamiferos AS(SELECT registo FROM animais
NATURAL INNER JOIN classe_bio
WHERE classe LIKE 'mamíferos'),
mamiferosperplace AS(SELECT registo_local, COUNT(animais) AS num
FROM espacos JOIN animais
ON espacos.registo_local=animais.local
JOIN mamiferos
ON animais.registo=mamiferos.registo
WHERE espacos.clima LIKE '%húmido%'
GROUP BY registo_local)
SELECT registo_local FROM espacos
WHERE registo_local=
(SELECT registo_local
FROM mamiferosperplace
WHERE num=(SELECT MAX(num)
FROM mamiferosperplace));
```

registo_local

A3

Exercício 10. o)

Para cada tratador indique o número de mamíferos por que é responsável?

```
WITH mamiferos AS(SELECT registo FROM animais
NATURAL INNER JOIN classe_bio
WHERE classe LIKE 'mamíferos'),
mamiferospertratador AS (SELECT nif,registo
FROM tratador NATURAL INNER JOIN mamiferos)
SELECT nome_func, COUNT(mamiferospertratador.nif)
AS mamifpertrat
FROM funcionario JOIN mamiferospertratador
ON funcionario.nif=mamiferospertratador.nif
GROUP BY nome_func;
```

nome_funcionario	mamifpertrat
Manuel Santos	8
Joaquim Silva	7

Exercício 10. p)

Indique o nome dos animais que já foram tratados por todos os veterinários?

```
WITH constPedroVale AS (SELECT registo FROM consultas
NATURAL INNER JOIN veterinarios
NATURAL INNER JOIN funcionario
WHERE funcionario.nome_func LIKE 'Pedro Vale'),
constIsabelSoares AS (Select registo FROM consultas
NATURAL INNER JOIN veterinarios
NATURAL INNER JOIN funcionario
WHERE funcionario.nome_func LIKE 'Isabel Soares' )
SELECT DISTINCT nome FROM animais
NATURAL INNER JOIN constPedroVale
NATURAL INNER JOIN constIsabelSoares;
```

nome
Tapi
Zula