



Base de dados

Relatório do Trabalho Prático

Base de dados Jardim Zoológico

3º semestre 2016/2017

Docente: Irene Rodrigues

Alunos: João Dias nº 35476

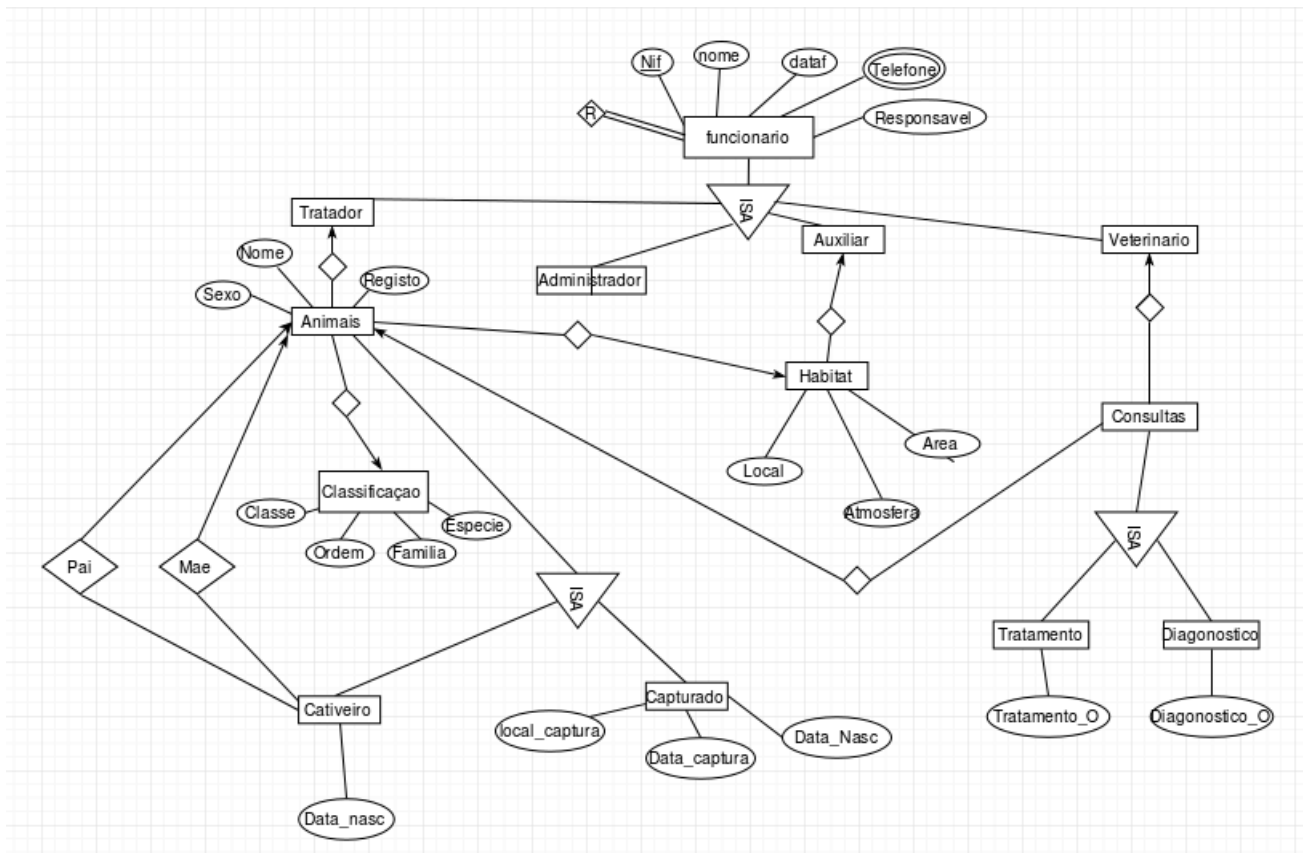
Eduardo Romão nº 35477

Introdução

No âmbito da unidade curricular de Base de dados, incorporada no programa de Licenciatura em Engenharia Informática, segundo ano, semestre ímpar, foi-nos pedido que elaborássemos um trabalho prático como instrumento de avaliação. Este trabalho consiste na criação de uma base de dados que armazene toda a informação de um jardim zoológico.

Exercício 1

Diagrama Entidades-Relação.



Exercício 2

Tabelas modelo E-R.

Funcionário (nif, nome, dataf)

Funcionário_telefone (nif, telefone)

Funcionário_responsavel (nif, responsavel)

Administrativo (nif)

Tratador (nif)

Auxiliar (nif)

Veterinário (nif)

Animais (registro, nome, sexo, nif, espécie, local_habitat)

Classificação (espécie, família, ordem, classe)

Cativeiro (registro, data_nascimento, registro_mae, registro_pai)

Capturado (registro, local_captura, data_nascimento, data_captura)

Habitat (local_habitat, atmosfera, área)

Consulta (nif, data_hora, local_consulta, registro)

Tratamento (nif, data_hora, local_consulta, tratamento_O)

Diagnostico (nif, data_hora, local_consulta, diagnostico_O)

Auxiliar_habitat (local_habitat, nif)

Exercício 3

Dependências funcionais.

- Animal → tratador = registro, nome, sexo, especie, nif, local_habitat
→ nif, registro
- Espécie → família
- Família → ordem
- Ordem → classe
- Registro → nome, sexo
- Animal → classificação = registro, nome, sexo, especie, nif,
local_habitat → ordem, classe, espécie, família
- Consulta → veterinário = data_hora, local_consulta → nif,
data_hora, local_consulta
- Capturado → animal = registro, local_captura, data_nascimento,
data_captura → registro, nome, sexo
- Cativo → animal = registro, data_nascimento → registro, nome,
sexo
- Cativo → cativo_mae = registro, data_nascimento → registro,
registro_mae
- Cativo → cativo_pai = registro, data_nascimento → registro,
registro_pai
- Tratamento → consulta = data_hora, local_consulta, tratamento_o →
data_hora, local_consulta
- Diagnostico → consulta = data_hora, local_consulta,
diagnostico_o → data_hora, local_consulta
- Animal → habitat = registro, nome, sexo → local_habitat, atmosfera,
área
- Consulta → animal = data_hora, local_consulta → registro, nome,
sexo

Exercício 4

Cobertura canónica.

Junção de todos os lados esquerdos das dependências

Registo, nome, sexo → nif, registo, ordem, classe, espécie, família,
local_habitat, atmosfera, área
Espécie → família
Família → ordem
Ordem → classe
Registo → nome, sexo
Data_hora, local_consulta → nif, data_hora, local_consulta, registo,
nome, sexo
Registo, local_captura, data_nascimento, data_captura → registo,
nome sexo
Registo, data_nascimento → registo, registo_mae, registo_pai
Data_hora, local_consulta, tratamento_O → data_hora,
local_consulta
Data_hora, local_consulta, diagnostico_O → data_hora,
local_consulta

Atributos extra lado direito

Registo → nif, espécie, local_habitat, atmosfera, área
Data_hora, local_consulta → nif, registo
local_captura, data_nascimento(captura), data_captura → registo
data_nascimento(cativeiro) → registo, registo_mae, registo_pai

Cobertura final

Registo → nif, espécie, local_habitat, atmosfera, área

Espécie → familia

Família → ordem

Ordem → classe

Registo → nome, sexo

Data_hora, local_consulta → nif, registo

local_captura, data_nascimento(captura), data_captura → registo

data_nascimento(cativeiro) → registo, registo_mae, registo_pai

Data_hora, local_consulta, tratamento_O → data_hora,

local_consulta

Data_hora, local_consulta, diagnostico_O → data_hora,

local_consulta

Exercício 5

Base de dados na forma normal de Boyce Codd.

Podemos concluir que esta base de dados já está na forma normal de Boyce Codd graças às dependências nela contida.

Relembramos que para a base de dados estar na forma normal de Boyce Codd todas as suas tabelas (relações) têm de estar na respetiva forma normal. Para isso todas as dependências dessas mesmas tabelas tem de ser trivias ou o fecho do lado esquerdo tem de ser superchave.

Como as dependências que temos estão todas contidas em mais que uma relação em simultâneo nunca podemos fazer o fecho de modo a obtermos uma superchave de uma só tabela, por isso concluimos que já esta na forma normal de Boyce Codd.

Exercício 6

Base de dados na 3ª forma normal.

Como a base de dados preserva as dependências não precisamos de colocar na 3ª forma normal.

Exercício 7

Chaves primárias, candidatas e estrangeiras de cada relação.

Relações	Chaves Primárias	Chaves Candidatas	Chaves Estrangeiras
Funcionario	Nif	Nif, nome, dataf	-/-
Funcionario_telefone	Nif, telefone	Nif, telefone	Nif
Funcionario_responsavel	Nif	Nif, responsavel	Nif
Administrativo	Nif	Nif	Nif
Tratador	Nif	Nif	Nif
Auxiliar	Nif	Nif	Nif
Veterinário	Nif	Nif	Nif
Animais	Registo	Registo, Nif	Nif, especie, loca_habitat
Classificação	espécie	especie	-/-
Cativeiro	Registo	Registo, data_nascimento	Registo, registo_mae, registo_pai
Capturado	Registo	Registo	registo
Habitat	local_habitat	local_habitat	-/-
Consulta	Nif, data_hora	Nif, data_hora	Nif, registo
Tratamento	Nif, data_hora	Nif, data_hora	Nif
Diagnostico	Nif, data_hora	Nif, data_hora	Nif
Auxiliar_habitat	local_habitat, Nif	local_habitat, Nif	Local_habitat, Nif

Exercício 8

Comandos SQL para a criação de tabelas.

Ficheiro em anexo Zoo_tabelas.sql

Exercício 9

Expressões SQL para inserir informação na base de dados.

Ficheiro em anexo Zoo_insert.sql

Exercício 10

(a) Em que locais do zoo se podem visitar aves?

- SQL:

```
select distinct(Local_habitat)
  from classificacao natural inner join animais
 where Classe='Aves';
```

(b) Em que locais do zoo não há carnívoros?

- SQL:

```
(select Local_habitat
  from habitat)
except
(select distinct(Local_habitat)
  from classificacao natural inner join animais
   where Ordem='Carnivoros');
```

(c) Indique os irmãos da Kilu (inclusive meios irmaos).

- SQL:

```
(select distinct(Registo)
 from (select Registo_mae, Registo_pai
       from cativoiro natural inner join animais
       where Nome='Kilu') as tab1, cativoiro
 where cativoiro.Registo_mae=tab1.Registo_mae or
       cativoiro.Registo_pai=tab1.Registo_pai)
except
(select Registo
 from animais
 where Nome='Kilu');
```

(d) Indique os telefones do tratador responsável pela Kata

- SQL:

```
select Telefone
from animais natural inner join tratador,
     funcionario natural inner join funcionario_telefone
where tratador.Nif=funcionario.Nif and animais.Nome='Kata';
```

(e) Indique os telefones do responsável pelo auxiliar responsável pelo local onde está a Kata.

- SQL:

```
select telefone
from funcionario_telefone
where funcionario_telefone.nif = (select responsavel
                                   from animais natural inner join habitat natural inner join
                                   funcionario_responsavel
                                   where animais.nome='Kata' and animais.nif =
                                   funcionario_responsavel.nif);
```

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

- SQL:

```
select tratamento.data_hora, tratamento_O
from animais ,consulta natural inner join tratamento
where animais.Registo = consulta.Registo and animais.Nome='Mali';
```

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

- SQL:

```
select funcionario.Nome  
from classificacao natural inner join animais, consulta natural inner join  
diagnostico natural inner join veterinario, funcionario  
where animais.Registo = consulta.Registo and Diagnostico_O='Grávida'  
and ordem='Carnivoros' and veterinario.Nif=funcionario.Nif;
```

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

- SQL:

```
select familia, count(Registo)  
from classificacao natural inner join animais  
where ordem='Artiodáctilos'  
group by familia;
```

- (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.

- SQL:

```
(select A.Nome, B.Nome
from animais as A, animais as B natural inner join cativo as C
where A.Registo!=B.Registo and A.Especie=B.Especie and
A.Registo!=C.Registo_mae and
      A.Registo!=C.Registo_pai and A.Sexo!=B.Sexo)
union
(select A.Nome, B.Nome
from animais as A, animais as B
where A.Registo!=B.Registo and A.Especie=B.Especie and
A.Sexo!=B.Sexo)
```

- (j) Qual é a ordem com mais animais no zoo?

- SQL:

```
select Ordem
from (select Ordem, count(Registo) as N
      from classificacao natural inner join animais
      group by Ordem) as tab1,
      (select max(tab2.N) as M
      from (select Ordem, count(Registo) as N
            from classificacao natural inner join animais
            group by Ordem) as tab2)as tab3
where tab1.N = tab3.M;
```


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

- SQL:

```
select Ordem
from (select Ordem, count(Data_hora) as N
      from classificacao natural inner join animais, consulta
      where animais.Registo = consulta.Registo
      group by Ordem) as tab
where tab.N>5;
```

(l) Indique o numero de animais nascidos em cativeiro.

- SQL:

```
select count(Data_nascimento)
from cativeiro;
```

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

- SQL:

```
select Nome, Especie
from((select Registo, extract(year from Data_nascimento)
      from cativeiro)
     union
     (select Registo, extract(year from Data_nascimento)
      from capturado)) as tab,
(select min(B.date_part) as Q
 from ((select Registo, extract(year from Data_nascimento)
        from cativeiro)
      union
      (select Registo, extract(year from Data_nascimento)
       from capturado)) as B) as tab2, animais
      natural inner join classificacao
where tab.date_part=tab2.Q and animais.Registo=tab.Registo;
```

(n) Qual é o local húmido com mais mamíferos?

- SQL:

```
select Local_habitat
from(select Local_habitat, count(Registo) as C
      from habitat natural inner join animais natural inner join
      classificacao
      where Atmosfera='Quente e húmida' and classificacao.classe =
      'Mamiferos'
      group by Local_habitat) as tab,

      (select max(tab2.C) as M
      from(select Local_habitat, count(Registo) as C
            from habitat natural inner join animais natural inner join
            classificacao
            where Atmosfera='Quente e húmida' and classificacao.classe =
            'Mamiferos'
            group by Local_habitat) as tab2)as tab3
where tab.C=tab3.M;
```

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

- SQL:

```
select funcionario.Nome, count(Registo)
from classificacao natural inner join animais natural inner join tratador,
funcionario
where Classe='Mamiferos' and tratador.Nif=funcionario.Nif
group by funcionario.Nome;
```

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

- SQL:

```
select Nome
from (select count(Nif) as C
      from veterinario) as tab,
      (select consulta.Registo, count(distinct(consulta.Nif)) as N
      from animais, consulta
      where animais.Registo = consulta.Registo
      group by consulta.Registo) as tab2, animais
where tab.C=tab2.N and animais.Registo=tab2.Registo;
```