

Projeto de Bases de Dados - Parte 3

Maria Beatriz Venceslau - 93734 - 14 horas

Helena Teixeira - 93720 - 14 horas

Maria Joana Lobo - 93736 - 14 horas

***Todo o esforço foi em simultâneo por zoom.**

G98 Turno: Terça-feira 8:30

Docente: Carlota Dias

Comandos de criação da Base de Dados:

```
drop table analise cascade;  
drop table regioao cascade;  
drop table concelho cascade;  
drop table instituicao cascade;  
drop table medico cascade;  
drop table consulta cascade;  
drop table prescricao cascade;  
drop table venda_farmacia cascade;  
drop table prescricao_venda cascade;
```

create table regioao

```
(num_regiao int not null unique,  
nome varchar(80) not null,  
num_habitantes int not null,  
constraint pk_regiao primary key(num_regiao),  
constraint chk_nome check(nome in ('Norte','Centro','Lisboa','Alentejo','Algarve')));
```

*****Esta última restrição poderia ser omitida sendo que a tabela região é read-only.*****

create table concelho

```
(num_concelho int not null,  
num_regiao int not null,  
nome varchar(80) not null,  
num_habitantes int not null,  
constraint pk_concelho primary key(num_concelho,num_regiao),  
constraint fk_regiao foreign key(num_regiao) references regioao(num_regiao));
```

*****Como a tabela concelho é read-only a restrição RI-concelho-1: nome = {concelhos de portugal continental} não foi explicitamente implementada no schema mas no populate apenas inserimos concelhos válidos presentes no portal das finanças. *****

create table instituicao

```
(nome varchar(80) not null unique,  
tipo varchar(80) not null,  
num_regiao int not null,  
num_concelho int not null,  
constraint pk_instituicao primary key(nome),  
constraint fk_instituicao_concelho foreign key(num_concelho,num_regiao) references  
concelho(num_concelho,num_regiao),  
constraint chk_tipo check(tipo in ('farmacia', 'laboratorio', 'clinica', 'hospital')));
```

create table medico

```
(num_cedula int not null unique,  
nome varchar(80) not null,  
especialidade varchar(80),  
constraint pk_medico primary key(num_cedula));
```

create table consulta

```
(num_cedula int not null,  
num_doente int not null,  
data date not null,  
nome_instituicao varchar(80) not null,  
constraint pk_consulta primary key(num_cedula, num_doente, data),  
constraint fk_consulta_cedula foreign key(num_cedula) references medico(num_cedula),  
constraint fk_consulta_instituicao foreign key(nome_instituicao) references instituicao(nome),  
constraint chk_saturday check(extract (dow from data) < 6),  
constraint chk_sunday check(extract (dow from data) > 0),  
constraint uc_consulta unique(num_doente, data, nome_instituicao));
```

create table prescricao

```
(num_cedula int not null,  
num_doente int not null,  
data date not null,  
substancia varchar(80) not null,  
quant int not null,  
constraint pk_prescricao primary key(num_cedula,num_doente,data,substancia),  
constraint fk_prescricao_cedula foreign key(num_cedula,num_doente,data)  
references consulta(num_cedula,num_doente,data));
```

create table analise

```
(num_analise int not null unique,  
especialidade varchar(80) not null,  
num_cedula int,  
num_doente int,  
data date,  
data_registro date not null,  
nome varchar(80) not null,  
quant int not null,  
inst varchar(80) not null,  
constraint pk_analise primary key(num_analise),  
constraint fk_analise_cedula foreign key(num_cedula,num_doente,data)  
references consulta(num_cedula,num_doente,data),  
constraint fk_analise_inst foreign key(inst) references instituicao(nome));
```

RI: a consulta associada pode estar omissa; não estando, a especialidade da consulta tem de ser igual à do médico

*****Se a consulta for omissa significa que pelo menos um dos atributos número de cédula, número de doente e data encontra-se null. Caso contrário, para saber a especialidade da análise teríamos aceder à tabela do médico com o respectivo número de cédula e retornar a sua especialidade. Isto é eventualmente realizado através de triggers/procedures. Como não se encontra no âmbito desta entrega realizar este tipo de restrições, ao preencher o populate tivemos o cuidado de fazer esta atribuição corretamente*****

create table venda_farmacia

```
(num_venda int not null unique,  
data_registo date not null,  
substancia varchar(80) not null,  
quant int not null,  
preco float(24) not null,  
inst varchar(80) not null,  
constraint pk_venda_farmacia primary key(num_venda),  
constraint fk_num_venda foreign key(inst) references instituicao(nome));
```

create table prescricao_venda

```
(num_cedula int not null,  
num_doente int not null,  
data date not null,  
substancia varchar(80) not null,  
num_venda int not null,  
constraint pk_prescricao_venda primary key(num_cedula,num_doente,substancia,data,  
num_venda),  
constraint fk_pres_venda_venda foreign key(num_venda)  
references venda_farmacia(num_venda),  
constraint fk_pres_venda_cedula foreign key(num_cedula,num_doente,data,substancia)  
references prescricao(num_cedula,num_doente,data,substancia));
```

Consultas em SQL:

1. --(falta por no populate a data corrente ou alterar para as no comentário)

```
select num_concelho, num_regiao
from venda_farmacia natural join instituicao
where data_registo = current_date      --(dia '2020-11-15')
and venda_farmacia.inst = instituicao.nome
group by num_concelho, num_regiao
having sum(quant) >= all
    (select sum(quant)
     from venda_farmacia natural join instituicao
     where data_registo = current_date  --(dia '2020-11-15')
     and venda_farmacia.inst = instituicao.nome
     group by num_concelho, num_regiao);
```

- 2.

```
with aux as (select count(*) as counter, num_regiao, num_cedula
from instituicao natural join
    (select *
     from prescricao natural join consulta
     where data >= '2019-01-01' and data <= '2019-06-30') as s
where s.nome_instituicao = instituicao.nome
group by num_regiao, num_cedula)
```

```
select a.num_regiao, a.num_cedula
from aux a inner join
    (select num_regiao, max(counter) counter
     from aux b
     group by num_regiao)
b on a.num_regiao = b.num_regiao
and a.counter = b.counter;
```

3.

```
with tabela as (select num_cedula, substancia, num_concelho, nome
                  from prescricao_venda natural join instituicao natural join venda_farmacia
                  where instituicao.nome = venda_farmacia.inst
                  and prescricao_venda.substancia = 'aspirina'
                  and extract(year from prescricao_venda.data) = ( select extract( year from
current_date))
                  and prescricao_venda.data < current_date
                  and instituicao.tipo = 'farmacia'
                  and num_concelho = 4 and num_regiao = 0),
array_farmacias as (select array_agg ( nome) as nome from instituicao where tipo =
'farmacia' and num_concelho = 4 and num_regiao = 0 ),
array_por_ced as (select tabela.num_cedula, array_agg ( nome) farma from tabela
                  group by tabela.num_cedula)

select num_cedula
from array_por_ced, array_farmacias
where array_por_ced.farma <@ array_farmacias.nome
and array_por_ced.farma @> array_farmacias.nome;
```

4. --(nao existem analises nem prescricoes em dezembro)

```
with fez_analise as (select * from analise
                     where extract(month from analise.data_registo) = ( select extract( month from
current_date))
                     and analise.data_registo <= current_date),
faz_prescricao as (select * from prescricao_venda
                   where extract(month from prescricao_venda.data) = ( select extract(
month from current_date))
                   and prescricao_venda.data <= current_date)
select num_doente from fez_analise except select num_doente from faz_prescricao;
```