

Projeto de Bases de Dados

Parte 3

Francisco Pereira, 76196 – 4h (33%)

Frederico Delaere, 79674 – 4h (33%)

Pedro Mela, 78876 – 4h (33%)

Grupo 45, turno de Sexta-feira (8:00)

Professor Paulo Carreira

2 de Dezembro de 2017

1 Criação da Base de Dados

```
drop table reposicao cascade;
drop table evento_reposicao cascade;
drop table planograma cascade;
drop table prateleira cascade;
drop table corredor cascade;
drop table fornece_sec cascade;
drop table produto cascade;
drop table fornecedor cascade;
drop table constituída cascade;
drop table super_categoria cascade;
drop table categoria_simples cascade;
drop table categoria cascade;

create table categoria(
    nome    varchar(80) not null unique,

    constraint pk_categoria primary key(nome)
);

create table categoria_simples(
    nome varchar(80) not null unique,

    constraint pk_categoria_simples primary key(nome),
    constraint fk_categoria_simples_categoria foreign key(nome)
        references categoria(nome) ON DELETE CASCADE
);

create table super_categoria(
    nome varchar(80) not null unique,

    constraint pk_super_categoria primary key(nome),
    constraint fk_super_categoria_categoria foreign key(nome)
        references categoria(nome) ON DELETE CASCADE
);

create table constituída(
    super_categoria    varchar(80) not null,
    categoria          varchar(80) not null,

    constraint pk_constituída primary key(super_categoria, categoria),
```

```

        constraint fk_constituída_super_categoria foreign key(super_categoria)
            references super_categoria(nome) ON DELETE CASCADE,
        constraint fk_constituída_categoria foreign key(categoria)
            references categoria(nome) ON DELETE CASCADE
    );

create table fornecedor(
    nif      char(9) not null unique,
    nome     varchar(80) not null,

    constraint pk_fornecedor primary key(nif)
);

create table produto(
    ean              char(13) not null unique,
    design           varchar(255) not null,
    categoria        varchar(80) not null,
    forn_primario    varchar(80) not null,
    data_produto     date not null,

    constraint pk_produto primary key(ean),
    constraint fk_produto_categoria foreign key(categoria)
        references categoria(nome) ON DELETE CASCADE,
    constraint fk_produto_fornecedor foreign key(forn_primario)
        references fornecedor(nif) ON DELETE CASCADE
);

create table fornece_sec(
    nif char(9) not null,
    ean char(13) not null,

    constraint pk_fornece_sec primary key(nif, ean),
    constraint fk_fornece_sec_fornecedor foreign key(nif)
        references fornecedor(nif) ON DELETE CASCADE,
    constraint fk_fornece_sec_produto foreign key(ean)
        references produto(ean) ON DELETE CASCADE
);

create table corredor(
    nro          numeric(10, 0) not null unique,
    largura      numeric(10, 0) not null,

    constraint pk_corredor primary key(nro)
);

```

```

);

create table prateleira(
    nro      numeric(10, 0) not null,
    lado     varchar(8) not null,
    altura   varchar(8) not null,

    constraint pk_prateleira primary key(nro, lado, altura),
    constraint fk_prateleira_corredor foreign key(nro)
        references corredor(nro) ON DELETE CASCADE
);

create table planograma(
    ean          char(13) not null,
    nro          numeric(10, 0) not null,
    lado         varchar(8) not null,
    altura       varchar(8) not null,
    face         int not null,
    unidades     int not null,
    loc          SERIAL,

    constraint pk_planograma primary key(ean, nro, lado, altura),
    constraint fk_planograma_produto foreign key(ean)
        references produto(ean) ON DELETE CASCADE,
    constraint fk_planograma_prateleira foreign key(nro, lado, altura)
        references prateleira(nro, lado , altura) ON DELETE CASCADE
);

create table evento_reposicao(
    operador varchar(80) not null,
    instante timestamp not null,

    constraint pk_evento_reposicao primary key(operador, instante)
);

create table reposicao(
    ean          char(13) not null,
    nro          numeric(10, 0) not null,
    lado         varchar(8) not null,
    altura       varchar(8) not null,
    operador     varchar(80) not null,
    instante     timestamp not null,
    unidades     int not null,

```

```

constraint pk_reposicao primary key(ean, nro, lado, altura, operador, instante),
constraint fk_reposicao_planograma foreign key(ean, nro, lado, altura)
    references planograma(ean, nro, lado , altura) ON DELETE CASCADE,
constraint fk_reposicao_evento_reposicao foreign key(operador, instante)
    references evento_reposicao(operador, instante) ON DELETE CASCADE
);

```

2 SQL

- a) Qual o nome do fornecedor que forneceu o maior número de categorias? Note que pode ser mais do que um fornecedor.

```

select nome from (
    select nome, categoria from fornece_sec
    natural inner join produto
    natural inner join fornecedor
    group by nome,categoria

    UNION ALL

    select nome, categoria from fornecedor
    inner join produto on fornecedor.nif = produto.forn_primario
    group by nome,categoria
) as A
group by nome
having count(nome) >= all (
    select count(nome) from (
        select nome, categoria from fornece_sec
        natural inner join produto
        natural inner join fornecedor

        UNION ALL

        select nome, categoria from fornecedor
        inner join produto on fornecedor.nif = produto.forn_primario
    ) as B
    group by nome
);

```

- b) Quais os fornecedores primários (nome e nif) que forneceram produtos de todas as categorias simples?

```

select nome, nif from produto inner join fornecedor
on produto.forn_primario = fornecedor.nif
where categoria in ( select nome from categoria_simples )
group by nome, nif
having count(nome) = (
    select count(*) from categoria_simples
);

```

c) Quais os produtos (ean) que nunca foram repostos?

```

select ean from produto where ean not in (
    select ean from reposicao natural inner join produto
);

```

d) Quais os produtos (ean) com um número de fornecedores secundários superior a 10?

```

select ean from fornece_sec group by ean having count(*) > 10;

```

e) Quais os produtos (ean) que foram repostos sempre pelo mesmo operador?

```

select ean from reposicao group by ean having count(operador) = 1;

```

3 Desenvolvimento da Aplicação

A aplicação pode ser encontrada em <http://web.tecnico.ulisboa.pt/~ist178876/BD/proj2017/>.

A página começa em *index.php* onde são apresentadas 4 opções:

1. Categorias

ver_categoria.php – apresenta as categorias a todos os níveis de profundidade

ver_categoria.php?categoria= – por exemplo mostra todas as categorias que não pertencem a nenhuma super_categoria

ver_categoria.php?categoria=Carnes – apresenta as categorias da super_categoria Carnes

2. Produtos

produtos.php – permite ver todos os produtos inseridos. É possível através desta página mudar designações de Produtos e ver os fornecedores secundários(clicar Ver no produto em questão) e também remover produtos.

3. Reposição

reposição.php – apresenta inicialmente uma tabela com todos os eventos de reposição. É possível selecionar um produto em questão e filtrar(clicando *Submit*).

4. Fornecedores

fornecedores.php – apresenta todos os fornecedores e dá-nos a possibilidade de adicionar ou remover.

Existem basicamente 3 tipos de ficheiros que devemos referir: *adicionar_*.php*, *update_*.php* e *remover_*.php*, onde * pode ser produto, fornecedor ou categoria. *adicionar_*.php* é sempre um *form* que recebe os dados para a criação de um novo * (produto, fornecedor ou categoria). Esse *form* irá fazer *submit*(POST) dos dados para *update_*.php* onde são feitos os *inserts* na base de dados. *remover_*.php* remove uma instância de *.