

# Projeto de Base de Dados, Parte 3

1º Semestre 2017-2018

Professor André Vasconcelos



## **TURNO BD81795L09 (Sexta-Feira 8h30m) - Grupo 6**

83405 - João Neves (33% - 6 horas)

84710 - Diogo Vilela (33% - 6 horas)

84711 - Diogo Redin (33% - 6 horas)

# Aplicação Web

A aplicação Web desenvolvido em PHP é constituída por duas páginas principais:

**products.php** - Nesta página temos listados todos os produtos e podemos adicionar / remover produtos, conjuntamente com os seus fornecedores, assim como alterar a sua designação.

- **addProduct.php** - É executada quando o utilizador submete o formulário para criar um novo produto. São passados para este ficheiro os dados do novo produto e dos seus fornecedores.
- **removeProduct.php** - É executada quando na tabela de produtos o utilizador clica no link “Remover”. É passado como parâmetro para este ficheiro o EAN do produto a remover.
- **changeProductDesignation.php** - É executada quando o utilizador preenche o campo da designação de um produto e clica no botão “Update”. São passado como parâmetros o EAN do produto e a nova designação introduzida.
- **listRestock.php** - É executada quando o utilizador escolhe um produto a partir do menu drop-down na página. Lista os eventos de reposição de um dado produto.

**categories.php** - Nesta página temos listados todas as categorias e podemos adicionar / remover categorias com ou sem super-categoria.

- **addCategory.php** - É executada quando o utilizador submete o formulário para criar uma nova categoria. São passados para este ficheiro, a partir do categories.php, os dados da categoria. O utilizador pode escolher como super categoria uma categoria já existente na caixa de seleção ou preencher os campos para criar uma nova. Também pode optar por não associar uma super categoria. O nome da categoria simples e da super categoria são passados como parâmetros para este ficheiro. Apenas é permitida a adição de uma super categoria desta forma porque uma super categoria tem que ter sempre associada uma categoria simples (RI-RE3).
- **removeCategory.php** - É executada quando na tabela de categorias o utilizador clica no link “Remover”. É passado como parâmetro para este ficheiro o Nome da categoria a remover.
- **listCat.php** - É executada quando o utilizador clica no link “List-Subcategories” na tabela de super categorias. São passado como parâmetros o nome da super categoria.

# Consultas 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 DISTINCT categoria, fornecedor as nif FROM produto
   UNION
   SELECT DISTINCT categoria, nif FROM fornece_sec NATURAL JOIN produto) as cats NATURAL JOIN
fornecedor
GROUP BY nif, nome
HAVING count(categoria) >= ALL(
  SELECT count(categoria)
  FROM
    (SELECT DISTINCT categoria, fornecedor as nif FROM produto
     UNION
     SELECT DISTINCT categoria, nif FROM fornece_sec NATURAL JOIN produto) as cats
  GROUP BY nif);
```

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

```
SELECT f.nome, f.nif
FROM fornecedor f, produto p, categoria_simples c
WHERE p.categoria = c.nome and p.fornecedor = f.nif;
```

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

```
SELECT ean
FROM produto
EXCEPT
SELECT ean
FROM produto NATURAL JOIN reposicao;
```

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

```
SELECT ean
FROM fornece_sec
GROUP BY ean
HAVING count(nif) > 10;
```

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

```
SELECT ean
FROM reposicao
GROUP BY ean
HAVING count(operador) = 1;
```

# Criação da Base de Dados

```
-- Categoria
create table categoria (
    nome varchar(15) not null unique,
    constraint pk_categoria primary key (nome)
);

-- Categoria Simples
create table categoria_simples (
    nome varchar(15) not null unique,
    constraint fk_categoria foreign key (nome) references categoria(nome),
    constraint pk_categoria_simples primary key (nome)
);

-- Super Categoria
create table super_categoria (
    nome varchar(15) not null unique,
    constraint fk_categoria foreign key (nome) references categoria(nome),
    constraint pk_super_categoria primary key (nome)
);

-- Constituida
create table constituida (
    super_categoria varchar(15) not null,
    categoria varchar(15) not null,
    unique (super_categoria, categoria),
    constraint categoria_inception check (super_categoria != categoria),
    constraint fk_super_categoria foreign key (super_categoria) references super_categoria(nome),
    constraint fk_categoria foreign key (categoria) references categoria(nome),
    constraint pk_constituída primary key (super_categoria, categoria)
);

-- Fornecedor
create table fornecedor (
    nif numeric(9,0) not null unique,
    nome varchar(15) not null unique,
    constraint pk_fornecedor primary key (nif)
);
```

```

-- Produto
create table produto (
    ean numeric(20,0) not null unique,
    design varchar(50) not null,
    categoria varchar(15) not null,
    fornecedor numeric(9,0) not null,
    instante timestamp not null,
    constraint fk_categoria foreign key (categoria) references categoria(nome),
    constraint fk_fornecedor foreign key (fornecedor) references fornecedor(nif),
    constraint pk_produto primary key (ean)
);

-- Fornecedor Secundario
create table fornece_sec (
    nif numeric(9,0) not null,
    ean numeric(20,0) not null unique,
    constraint fk_fornecedor foreign key (nif) references fornecedor(nif),
    constraint fk_produto foreign key (ean) references produto(ean),
    constraint pk_fornece_sec primary key (nif, ean)
);

-- Corredor
create table corredor (
    nro numeric(4,0) not null unique,
    largura numeric(3,0) not null,
    constraint pk_corredor primary key (nro)
);

-- Prateleira
create table prateleira (
    nro numeric(4,0) not null,
    lado varchar(1) not null,
    altura numeric(3,0) not null,
    unique (nro, lado, altura),
    constraint fk_corredor foreign key (nro) references corredor(nro),
    constraint pk_prateleira primary key (nro)
);

```

```

-- Planograma
create table planograma (
    ean numeric(20,0) not null unique,
    nro numeric(4,0) not null,
    lado varchar(1) not null,
    altura numeric(3,0) not null,
    face varchar(1) not null,
    unidades numeric(4,0) not null,
    loc varchar(15) not null,
    constraint fk_produto foreign key (ean) references produto(ean),
    constraint fk_prateleira foreign key (nro, lado, altura) references prateleira(nro, lado, altura),
    constraint pk_planograma primary key (ean, nro, lado, altura)
);

-- Evento Reposicao
create table evento_reposicao (
    operador varchar(15) not null,
    instante timestamp not null,
    unique (operador, instante),
    constraint categoria_inception check (instante <= now()),
    constraint pk_evento_reposicao primary key (operador, instante)
);

-- Reposicao
create table reposicao (
    ean numeric(20,0) not null,
    nro numeric(4,0) not null,
    lado char(1) not null,
    altura numeric(3,0) not null,
    operador varchar(15) not null,
    instante timestamp not null,
    unidades numeric(4,0) not null,
    unique (ean, nro, lado, altura, operador, instante, unidades),
    constraint fk_planograma foreign key (ean, nro, lado, altura) references planograma(ean, nro, lado,
altura),
    constraint fk_evento_reposicao foreign key (operador, instante) references
evento_reposicao(operador, instante),
    constraint pk_reposicao primary key (ean, nro, lado, altura, operador, instante)
);

```