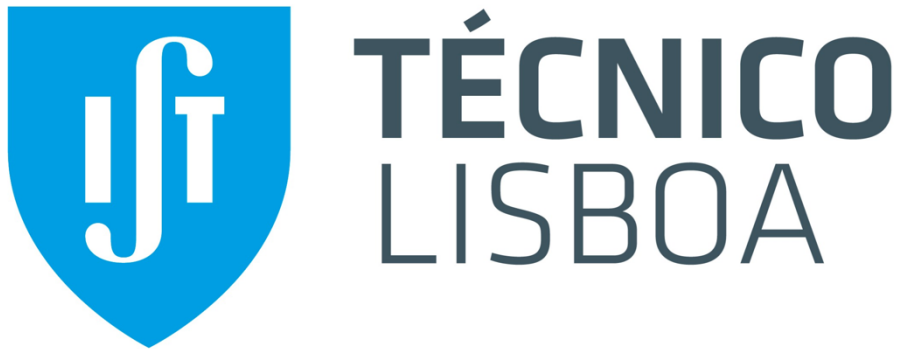


Projeto de Bases de Dados, 3º Entrega

2017/2018



Grupo 1

Membros:

- **84718** – Francisco Aguiar, 33% - 17 Horas
- **84719** – Gonçalo Marques, 33% - 17 Horas
- **84740** – Manuel Sousa, 34% - 18 Horas

Professor: Tomás Almeida e Silva Martins Alves

Turno: Quinta-Feira, 8:30 – 10:00 (BD81795L08)

Criação da Base de Dados

```
drop table Categoria cascade;
drop table Categoria_Simples cascade;
drop table Super_Categoria cascade;
drop table Constituida cascade;
drop table Fornecedor cascade;
drop table Produto cascade;
drop table Fornecedor_secundario cascade;
drop table Corredor cascade;
drop table Prateleira cascade;
drop table Planograma cascade;
drop table EventoReposicao cascade;
drop table Reposicao cascade;
```

```
CREATE TABLE Categoria(
    nome VARCHAR(50) NOT NULL,
    PRIMARY KEY(nome)
);
```

```
CREATE TABLE Categoria_Simples(
    nome VARCHAR(50) NOT NULL,
    PRIMARY KEY(nome),
    FOREIGN KEY (nome) REFERENCES Categoria ON
DELETE CASCADE
);
```

```
CREATE TABLE Super_Categoria(
    nome VARCHAR(50) NOT NULL,
    PRIMARY KEY(nome),
    FOREIGN KEY (nome) REFERENCES Categoria ON
DELETE CASCADE
);
```

```
CREATE TABLE Constituida(
    super_categoria VARCHAR(50) NOT NULL,
    categoria VARCHAR(50) NOT NULL,
```

```
    PRIMARY KEY(super_categoria,categoria),
    FOREIGN KEY (super_categoria) REFERENCES
Super_Categoria(nome) ON DELETE CASCADE,
    FOREIGN KEY (categoria) REFERENCES
Categoria(nome) ON DELETE CASCADE
);
```

```
CREATE TABLE Fornecedor(
    nif VARCHAR(9) NOT NULL,
    nome VARCHAR(25) NOT NULL,
    PRIMARY KEY (nif)
);
```

```
CREATE TABLE Produto(
    ean VARCHAR(25) NOT NULL,
    categoria VARCHAR(50) NOT NULL,
    forn_primario VARCHAR(9) NOT NULL,
    design VARCHAR(50) NOT NULL,
    data DATE NOT NULL,
    PRIMARY KEY (ean),
    FOREIGN KEY(categoria) REFERENCES Categoria
ON DELETE CASCADE,
    FOREIGN KEY(forn_primario) REFERENCES
Fornecedor
);
```

```
CREATE TABLE Fornecedor_secundario(
    nif VARCHAR(9) NOT NULL,
    ean VARCHAR(25) NOT NULL,
    FOREIGN KEY(nif) REFERENCES
Fornecedor(nif),
    FOREIGN KEY(ean) REFERENCES Produto(ean) ON
DELETE CASCADE
);
```

```
CREATE TABLE Corredor(
    nro INT NOT NULL,
    largura INT NOT NULL,
```

```

PRIMARY KEY(nro),
check(largura > 0),
check(nro > 0)
);

CREATE TABLE Prateleira(
    lado VARCHAR(8) NOT NULL,
    altura INT NOT NULL,
    nro INT NOT NULL,
    PRIMARY KEY(lado,altura,nro),
    FOREIGN KEY(nro) REFERENCES Corredor,
    CHECK(altura > 0)
);

CREATE TABLE Planograma(
    face INT NOT NULL,
    unidades INT NOT NULL,
    localizacao VARCHAR(10) NOT NULL,
    nro INT NOT NULL,
    ean VARCHAR(25) NOT NULL,
    lado VARCHAR(8) NOT NULL,
    altura INT NOT NULL,
    PRIMARY KEY(ean,nro,lado,altura),
    FOREIGN KEY(ean) REFERENCES Produto ON
DELETE CASCADE,
    FOREIGN KEY(lado,altura,nro) REFERENCES
Prateleira
);

```

```

CREATE TABLE EventoReposicao(
    operador VARCHAR(25) NOT NULL,
    instante TIMESTAMP NOT NULL,
    PRIMARY KEY(operador,instante)
);

```

```

CREATE TABLE Reposicao(
    unidades INT NOT NULL,
    ean VARCHAR(25) NOT NULL,

```

```

nro INT NOT NULL,
altura INT NOT NULL,
lado VARCHAR(8) NOT NULL,
operador VARCHAR(25) NOT NULL,
instante TIMESTAMP NOT NULL,
PRIMARY KEY(ean, nro, lado,
altura,operador,instante),
    FOREIGN KEY(ean) REFERENCES Produto ON
DELETE CASCADE,
    FOREIGN KEY(lado,altura,nro) REFERENCES
Prateleira,
    FOREIGN KEY(operador,instante) REFERENCES
EventoReposicao,
    CHECK(unidades > 0)
);

```

```

ALTER TABLE EventoReposicao
    ADD CONSTRAINT RI_EA3 CHECK(instante <=
CURRENT_TIMESTAMP);

```

```

ALTER TABLE Prateleira
    ADD CONSTRAINT TIPOLADO CHECK (lado =
'esquerdo' OR lado = 'direito');

```

SQL

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

```
WITH Produtores_por_produto as (  
  
    SELECT forn_primario as nif,ean FROM Produto  
  
    UNION  
  
    SELECT * FROM Fornecedor_secundario  
  
),Categoria_por_fornecedor as (  
  
    SELECT nif,categoria  
  
    FROM Produto, Produtores_por_produto  
  
    WHERE Produto.ean = Produtores_por_produto.ean  
  
), n_Categorias_por_fornecedor as (  
  
    SELECT nif, COUNT(categoria) as n_categorias  
  
    FROM Categoria_por_fornecedor  
  
    GROUP BY nif  
  
)  
  
SELECT nome  
  
FROM Fornecedor  
  
WHERE Fornecedor.nif IN (  
  
    SELECT nif  
  
    FROM n_Categorias_por_fornecedor  
  
    WHERE n_Categorias_por_fornecedor.n_categorias IN(  
  
        SELECT MAX(n_categorias) as max_cat  
  
        FROM n_Categorias_por_fornecedor  
  
    )  
  
);
```

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

```
WITH Produtor_primario AS (  
  
    SELECT forn_primario as nif, ean
```

```

FROM Produto
), Produtores_por_produto as (

SELECT nif,ean FROM Produtor_primario
UNION
SELECT * FROM Fornecedor_secundario

),Categorias_por_fornecedor as (

SELECT *
FROM Produto, Produtores_por_produto
WHERE Produto.ean = Produtores_por_produto.ean AND
Produto.categoria IN (SELECT nome FROM Categoria_Simples)

),n_Categorias_simples_por_fornecedor as(

SELECT nif, COUNT(distinct categoria) as n_categorias_simples
FROM Categorias_por_fornecedor
GROUP BY nif
)
SELECT Fornecedor.nif, nome
FROM Fornecedor
WHERE Fornecedor.nif IN (

SELECT nif
FROM n_Categorias_simples_por_fornecedor
WHERE n_Categorias_simples_por_fornecedor.n_categorias_simples IN (

SELECT COUNT(distinct nome) as n_categorias_simples
FROM Categoria_Simples)

);

```

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

```

SELECT ean from Produto where ean not in
(SELECT ean from Reposicao);

```

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

```

SELECT ean from Produto where ean in
(SELECT ean from Fornecedor_secundario group by ean having count(ean)>10);

```

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

```
SELECT ean from Reposicao group by ean having count(distinct operador) = 1;
```

PHP

A aplicação encontra-se estruturada por 3 níveis, o nível do menu principal, o nível de apresentação (pasta presentationTier) e o nível lógico de php (pasta applicationTier).

Nível de apresentação:

Não tem qualquer lógica, é apenas um nível de interface constituído por ficheiros HTML com as seguintes funções:

- **createProduct.html** → Contém a interface de inserção de um produto, com todos os campos necessários para inserir um produto na base de dados, e opção de adicionar 1 ou mais fornecedores secundários para o produto em questão, esta interface é responsável também por fazer o forwarding dos dados para a área de aplicação onde o ficheiro **produto.php** trata da parte lógica de inserção na BD.
- **deleteProduct.html** → Contém a interface de remoção de um produto, contém apenas um campo de inserção do ean que permite ao script no ficheiro da parte de aplicação, **deleteProd.php** fazer a remoção do produto da base de dados.
- **deleteCat.html** → Contém a interface de remoção de uma categoria, que é igual independentemente se estamos a remover uma categoria simples ou uma super categoria, pois apenas necessita do nome da categoria para que esta possa ser eliminada através do script **deleteCat.php**.
- **createCat.html** → Este ficheiro contém simplesmente 2 botões que permitem acesso a criação de uma categoria simples ou de uma super categoria, este ficheiro tem a função de redirecionar o utilizador para os seguintes ficheiros:
 - **createSuperCat.html** → Contém a interface com os campos necessários para inserir uma super categoria, como o nome desta, e a opção de adicionar 1 ou mais sub-categorias desta super-categoria, este ficheiro faz a ligação com o ficheiro **superCategoria.php** na área logica da aplicação que é responsável pela parte lógica da inserção na base de dados.
 - **CreateSimpleCat.html** → Contém a interface com o campo nome, que é o único campo necessário para que o ficheiro **categoria.php** possa fazer a inserção na base de dados.
- **listRep.html** → Contém a interface com o campo ean, que é o único necessário para o script no ficheiro **repositorao.php** listar todas as reposições do produto em questão.
- **listSubCat.html** → Contém a interface com o campo nome, que permite que o script **subCategorias.php** possa encontrar todas as subCategorias de uma dada categoria.
- **changeProd.html** → Contém a interface com os campos ean e designação que permitem ao script **designacao.php** alterar a designação do produto em questão para a nova descrição dada.

Nível lógico:

No nível lógico é onde ocorre a verificação dos dados fornecidos pelo utilizador e a sua inserção na base de dados, esta parte da aplicação é inteiramente constituída por ficheiros PHP com as seguintes funções:

- **produto.php** → responsável pela inserção dos dados do produto na base de dados, estes dados apenas são inseridos caso cumpram todas as restrições de integridade relacionadas com o produto, por exemplo se não for fornecido qualquer fornecedor secundário e/ou primário o produto não é inserido na base de dados e uma mensagem de erro é mostrada.
- **deleteProd.php** → responsável pela remoção de um produto da base de dados, esta remoção é feita apenas com o delete do produto e depois com todas as ocorrências do ean como foreign key estão marcadas na base de dados como ON DELETE CASCADE todas as instâncias que tenham esse ean vão ser eliminadas, como é o caso dos fornecedores secundários do produto em questão, que são eliminados da tabela fornecedores_secundarios (mas não da tabela fornecedores).
- **superCategoria.php** → responsável pela inserção de uma super categoria na base de dados, para que esta inserção seja possível têm de ser fornecidas outras categorias, simples ou super, novas ou existentes na base de dados que sejam sub-categorias desta categoria, caso contrário é mostrada uma mensagem de erro ao utilizador e a inserção é cancelada, a inserção é também cancelada caso esta já exista na base de dados, todas as novas sub categorias dadas são inseridas como categorias simples.
- **categoria.php** → responsável pela inserção de uma categoria simples na base de dados, este script simplesmente insere a categoria na tabela categoria_simples e na tabela categoria, caso esta nova categoria simples já exista na base de dados, é mostrada uma mensagem de erro ao utilizador e a inserção é cancelada.
- **deleteCat.php** → responsável pela remoção de um produto da base de dados, esta remoção é feita de modo similar com a de um produto, mas tem o cuidado de quando remove uma categoria que é sub-categoria de uma outra super-categoria, ver se essa categoria que era super, ficou agora sem sub-categorias e passa então a ser uma categoria-simples.
- **reposicao.php** → responsável pela listagem das reposições de um dado produto, caso o ean dado não corresponda a o de nenhum produto não são dados quaisquer resultados, e é imprimida uma resposta de erro.
- **subCategorias.php** → responsável pela listagem de todas as sub-categorias de uma data categoria e de todas as suas sub-categorias correspondentes.
- **designacao.php** → responsável por alterar a designação de um dado produto.