

## Projeto de Bases De Dados, Parte 3

## Relatório

Nº Grupo	Turno (Alameda/Taguspark, dia semana, e hora)	Professor	
34	Alameda, Quarta das 08:00 ás 09:30 LAB 10	André Vasconcelos	
Aluno:		Participação no projeto	
Nº	Nome	percentagem relativa de contribuição	Esforço em horas
81633	João Henriques	33,3333%	40
83434	Beatriz Toscano	33,3333%	40
83506	Mara Caldeira	33,3333%	40
Total:		100%	<u>120</u>

## Criação da base de dados

Para a criação da base de dados usámos os ficheiros "schema.sql" e "populate.sql", onde definimos a estrutura da nossa base de dados e populámos esta com dados, respetivamente.

```
Um exemplo de uma estrutura presente no ficheiro schema.sql, neste caso referente
a construção da tabela "constituída"
drop table if exists constituida cascade;
create table constituida(
      super categoria varchar(255) not null,
      categoria varchar(255) not null unique,
      primary key(super_categoria, categoria),
      foreign key(categoria) references categoria(nome) ON UPDATE cascade.
      foreign key(super categoria) references super categoria(nome) ON UPDATE
cascade.
      check(categoria != super_categoria)
);
E o seu respetivo populate
insert into constituida values ('Carne', 'Enlatados');
insert into constituida values ('Pronto-a-comer', 'Pizza');
insert into constituida values ('Higiene', 'Shampoo'):
insert into constituida values ('Enlatados', 'Salsichas');
<u>SQL</u>
-- A) Qual o nome do fornecedor que que forneceu o maior numero de categorias
-- seleciona o nome
select nome
from fornecedor
where nif = (select forn_primario --ve o nif que correponde ao count
       from produto
       group by forn primario
       having count(*) = ( select MAX(y.num) --ve qual e o maior count
                   from (select count(*) as num
                   from produto as x
                    group by forn_primario) y));
-- B) Quais os fornecedores primários(nome e nif) que forneceram produtos de todas
as categorias simples
select fornecedor.nome, fornecedor.nif
from categoria, produto, categoria simples, fornecedor
where categoria.nome = produto.categoria and categoria.nome =
categoria_simples.nome and fornecedor.nif = produto.forn_primario;
-- C) Quais os produtos(ean) que nunca foram repostos
select ean
from produto
where produto.ean NOT IN( select ean
                 from reposicao)
group by ean;
```

-- D) Quais os produtos(ean) com um número de fornecedores secundários superior a 10

select produto.ean from produto natural join fornece\_sec group by ean

having count(ean) > 10;

-- E) Quais os produtos(ean) que foram repostos sempre pelo mesmo operador select produto1.ean

from produto as produto1

where 1 = (select count(distinct evento\_reposicao.operador)

from evento\_reposicao natural join reposicao natural join produto as produto2

where produto1.ean = produto2.ean);

## Desenvolvimento da aplicação

Para facilitar o desenvolvimento da aplicação criamos um ficheiro "menulnicial.php", onde dá para aceder aos links refentes às 5 alíneas pedidas para esta entrega. Pode aceder a este mesmo ficheiro e testar a nossa aplicação em: http://web.ist.utl.pt/ist181633/BD/BD-3%C2%BA%20entrega/menulnicial.php.

Para a alínea a) usamos os ficheiros:

"categorias.php" que é onde o utilizador escolhe se quer adicionar ou remover uma categoria;

"addCategoria.php" que é onde o utilizador escolhe se pretende adicionar uma Super categoria, ou uma categoria simples;

"catSimples.php" e "superCat.php" que é o ficheiro onde o utilizador submete a sua nova categoria simples e super categoria respetivamente;

"updateSimples.php" e "updateSuper.php" é o ficheiro que faz update a base de dados com a nova categoria simples e super categoria respetivamente;

"delCategoria.php" é o ficheiro que permite ao utilizador selecionar uma categoria para apagar

"updateDel.php" é o ficheiro que faz update a base de dados eliminando a categoria selecionada

Para a alínea b) usamos os ficheiros:

"doProdutos.php" onde o utilizador pode escolher entre remover e adicionar um produto;

"produtosAdd.php" onde o utilizador escolhe os parâmetros do novo produto;

"up\_Add\_produto.php" ficheiro que atualiza a base de dados com o novo ficheiro;

"produto\_forn\_sec.php" permite ao utilizador adicionar fornecedores secundários ao produto;

"up\_prod\_forn\_sec.php" atualiza a base de dados com estes mesmos fornecedores:

"produtosRem.php" permite ao utilizador escolher o produto a remover;

"up rem produto.php" remove o produto da base de dados;

Para a alínea c) usamos os ficheiros:

"event\_rep.php" onde o utilizador escolhe o produto que pretende ver os eventos de reposição

"list\_produto.php" o ficheiro que mostra esses mesmos eventos;

Para a alínea d) usamos os ficheiros:

"changeProduto.php" o ficheiro onde o utilizador escolhe o produto a alterar e a sua nova designação;

"updateProduct.php" o ficheiro que atualiza a base de dados com a nova designação do produto;

Para a alínea e) usamos os ficheiros:

"listaSub.php" onde o utilizador escolhe a categoria que pretende;

"updateListSub.php" o ficheiro que mostra a sub categorias da categoria selecionada.

De um modo geral, quando o utilizador tem de selecionar alguma informação vinda da base de dados, usamos uma estrutura em forma de tabela com "radio buttons", este tipo de input permite que o utilizador escolha só um elemento da tabela. Ordenamos também os valores da tabela pela sua chave primária.

Em baixo segue como exemplo dois dos ficheiros php, concretamente o ficheiro "listaSub.php" e "updateListSub.php", respetivos a alinea e):

```
listaSub.php
<html>
      <head>
            <meta charset="utf-8">
      </head>
      <body>
            <h3>Categorias</h3>
            <form action="updateListSub.php" method="post">
                  <?php
                        try{
                              //ligacao a base de dados
                              $host = "db.ist.utl.pt";
                              $user ="ist181633";
                              $password = "projBD";
                              $dbname = $user;
                              db = new
PDO("pgsql:host=$host;dbname=$dbname", $user, $password);
                              $db->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE EXCEPTION):
                              //querie para ir a base de dados buscar todos os
nomes das super categorias
                              $sql = "SELECT nome
                                          FROM super_categoria
                                          ORDER BY nome;";
                              $result = $db->query($sql);
                              //tabela que mostra estes nomes
                              //a tabela tem radio buttons para o utilizador
escolher uma das categorias
                              echo("\n");
                              echo("\n");
```

```
echo("NOME\n");
                                    echo("\n");
                              echo("\n");
                              foreach($result as $row){
                                    echo("\n");
                                          echo("{row['nome']}\n");
                                          echo("<input type=\"radio\"
name=\"superCat\" value=\"{\fow['nome']}\">\n");
                                    echo("\n");
                              echo("");
                              db = null;
                        }
                        catch (PDOException $e){
                              echo("ERROR: {$e->getMessage()}");
                        }
                  ?>
                  <input type="submit" value="Submit"/>
            </form>
            <a href="menulnicial.php">Voltar ao Menu</a>
      </body>
</html>
updateListSub.php
</head>
      <body>
            <?php
      /*É funcao uma recursiva que imprime as subcategorias de todas as
categorias e
      é chamada recursivamente em cada subcategoria para também imprimir as
subcategorias dela,
      caso ela tenha*/
                  function printSub($row){
                        try{
           //ligacao a base de dados
           $host = "db.ist.utl.pt";
           $user ="ist181633":
           $password = "proiBD";
           $dbname = $user;
           $db = new PDO("pgsql:host=$host;dbname=$dbname", $user,
$password);
           $db->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
           //ve as categorias simples, que constituem a super categoria
selecionada
           $sql = "SELECT categoria
               FROM constituida
               WHERE super_categoria = '{$row}';";
           $db->query($sql);
```

```
$result = $db->query($sql);
           foreach ($result as $row){
                echo (" {$row['categoria']}");
                //chama a funcao outra vez para imprimir as sub categorias desta
categoria
                printSub($row['categoria']);
           }
         }
         catch (PDOException $e){
           echo("ERROR: {$e->qetMessage()}");
         }
                   $superCat = $_REQUEST['superCat'];
                   try{
         //ligacao a base de dados
                         $host = "db.ist.utl.pt";
                         $user ="ist181633";
                         $password = "projBD";
                         $dbname = $user;
                         $db = new PDO("pgsql:host=$host;dbname=$dbname",
$user, $password);
                         $db->setAttribute(PDO::ATTR ERRMODE,
PDO::ERRMODE_EXCEPTION);
         //ve as categorias simples, que constituem a super categoria selecionada
pelo utilizador
                         $sql = "SELECT categoria
                                      FROM constituida
                                      WHERE super_categoria = '$superCat';";
                         $db->query($sql);
                         \text{sesult} = \text{db->query($sql)};
                         echo("<b>Sub-Categorias:</b>\n");
                         foreach ($result as $row) {
              //imprime as categorias que constituem a super categoria
                                      echo (" {$row['categoria']}");
              //ve se estas categorias da super categoria selecionada tem sub-
categorias
                                      printSub($row['categoria']);
                         db = null;
                   catch (PDOException $e){
                         echo("ERROR: {$e->getMessage()}");
            ?>
             <a href="menulnicial.php">Voltar ao Menu</a>
      </body>
</html>
```