

Projeto BD - Parte 3

Turno BD2L16

Docente Daniel Mateus Gonçalves

Grupo 075

Número	Nome	Horas de trabalho	Percentagem Relativa
92559	Sofia Bonifácio		%
95530	Ana Sequeira		%
95574	Filipe Silva		%

Comentários

Tirando os índices, o restante do projeto está nos ficheiros incluídos no zip.

[Versão de trabalho \(link\)](#)

7. Índices

7.1

```
SELECT DISTINCT R.nome  
FROM retalhista R, responsavel_por RP  
WHERE R.tin = RP.tin and RP.nome_cat = 'Frutos'
```

Lista de retalhistas responsáveis por produtos da categoria Frutos.

Chave de procura composta, com índices de dispersão (dinâmica) nos atributos nome_cat e tin da tabela responsavel_por, irá otimizar a procura destes atributos na tabela, visto que são ambos precisos, um por cada igualdade (onde os índices de dispersão são eficientes).

A tabela retalhista já tem tin como chave primária, pelo que é desnecessário fazer um índice para esta.

```
CREATE INDEX idRP ON responsavel_por hash (nome_cat, tin);
```

7.2

```
SELECT T.nome, count(T.ean)  
FROM produto P, tem_categoria T  
WHERE P.cat = T.nome and P.desc like 'A%'  
GROUP BY T.nome
```

Categorias que contém um produto cuja descrição começa por A e respectivo número de produtos.

Tendo em conta que sabemos como começa a pesquisa no índice para P.desc, podemos usar um índice (ao contrário do caso de '%A', por exemplo).

Usando uma chave de procura composta, com índices btree nos atributos cat e desc da tabela produto, irá otimizar a procura destes atributos na tabela, visto que são ambos precisos, um por cada comparação. Não se poderá usar índices hash, visto que a operação like não é suportada nestes.

A tabela tem_categoria já tem nome como chave primária (conjunta com ean), pelo que é desnecessário fazer um índice para esta.

```
CREATE INDEX idP ON produto btree (cat, desc);
```