Questão 5 – Query Tuning

5.1)

A razão para o otimizador não encontrar um bom plano de execução prende-se que o otimizador não utiliza os índices presentes em expressões aritméticas, como o caso da expressão *(calories \* 100 /800).* A solução passa por simplificar a expressão ficando apenas o atributo *calories* do lado esquerdo da desigualdade.

|  |
| --- |
| SELECT name  FROM CHEESE  WHERE calories < 8 |

5.2)

O optimizador não irá tirar partido do indíce na sub-expressão *calories < 90* porque como os registos estão ordenados de acordo com o atributo *calories*, é apenas feito o scan a tabela enquanto se verificar a condição *calories < 90.*

A query sugerida é:

|  |
| --- |
| SELECT name  FROM CHEESE  WHERE calories < 40 AND calories <90 |

A razão é porque na expressão *calories > 40* é utilizado o índice para aceder a ao primeiro registo em que esta condição sem verifique e depois é feito um scan até a condição *calories < 90* já não se verificar.

5.3)

A query sugerida é:

|  |
| --- |
| SELECT type  FROM CHEESE  WHERE calories = 90  UNION ALL  SELECT type  FROM CHEESE  WHERE calories = 40 |

Na presença do operador OR alguns optimizadores nunca utilizam os índices contidos nessas expressões mas, ao utilizar duas expressões unidas pela keyword UNION, o otimizador utiliza os índices e realiza uma união dos resultados, não eliminando os resultados repetidos por causa da utilização da keyword ALL seguida de UNION, se não se utiliza-se a keyword ALL o SGBD procederia a uma eliminação dos resultados repetidos, mas visto que os conjuntos de resultados *calories = 90* e *calories = 40* são disjuntos, esta verificação apenas representaria um overhead para o sistema.

5.4)

Como nos valores retornados pelas query está contido o atributo-chave então todos os registos retornados são diferentes, logo é desnecessária a utilização do DISTINCT, pois apenas acarretaria um overhead ao verificar se existem registos iguais quando não existem.

A query rescrita é:

|  |
| --- |
| SELECT cheeseID, type  FROM CHEESE |

5.5)

A keyword HAVING deve ser reservada para propriedades agregadas dos grupos de registos, como não é o caso da query então é utilizado o WHERE para retornar os mesmos tuplos que retornaria a keyword HAVING.

A query sugerida é:

|  |
| --- |
| SELECT AVG(proteins)  FROM CHEESE  WHERE type = "Alverca"  GROUP BY producer |