

Complementos de Bases de Dados

2022/2023

Licenciatura em Engenharia Informática



Laboratório 3 – Índices e Monitorização

Objetivos:

- Utilização de índices Clustered e Non-Clustered;
- Visualização do impacto de queries através do Execution Plan.
- Ferramentas de Monitorização
 - Explorar o Execution Plan;
 - Utilização de ferramentas de monitorização.

Enunciado:

Descarregue os scripts facultados juntamente com este enunciado:

- Indices_sobreutilizacao.sql
- BTree.sql

Estes scripts disponibilizarão um conjunto de dados e instruções, onde assentará este enunciado.

ETAPA 1

1. Verifique se a coluna *CustomerID* da tabela *Customer* é IDENTITY.
2. Liste todos os índices existentes na tabela *Customer*, e identifique o seu tipo.
3. Os índices permitem otimizar determinadas queries que filtrem os resultados pelos valores de colunas. Daí os índices serem criados por colunas. Contudo, a existência de um índice cria um “overhead” na manutenção do índice aquando de operações DML. Portanto, a criação de índices sobre tabelas deve ser feita criteriosamente, com um balanço em mente do tipo de operações mais frequentes que serão feitas sobre a tabela.

Sabendo que a seletividade é o rácio entre os valores únicos de uma coluna sobre um conjunto de valores, crie um script que calcule a seletividade de todas as colunas da tabela *Customer*.

4. Quais as colunas candidatas à criação de índices tendo em conta os resultados obtidos na alínea anterior? Justifique a resposta.

ETAPA 2

1. Visualize as estatísticas e correspondente *Execution Plan* das seguintes instruções:

```
SET STATISTICS IO ON
```

```
SELECT c.LastName , c.FirstName  
FROM SalesLT.Customer c  
WHERE CustomerID=100
```

```
SELECT c.LastName , c.FirstName  
FROM SalesLT.Customer c  
WHERE c.phone ='979-555-0163'
```

Analise e comente qual a diferença no que diz respeito à utilização dos índices?

2. Crie um índice non-clustered com o nome *NONCI_phone* sobre a coluna *phone* e repita as queries anteriores forçando a utilização do índice anterior e explique o resultado.
3. Execute uma query que devolva todos os *phone* iniciados por “96”, guarde o resultado do *Execution Plan* e observe o seu resultado.
4. Proponha justificadamente e crie um índice que optimize as seguintes queries (potencialmente frequentes):

```
SELECT c.LastName , c.FirstName, c.EmailAddress  
FROM SalesLT.Customer c  
WHERE c.LastName LIKE 'A%'
```

```
SELECT c.LastName , c.FirstName, c.EmailAddress  
FROM SalesLT.Customer c  
ORDER BY c.LastName
```

ETAPA 3

1. Abra o script “Indices_sobreutilizacao_Customer.sql” e execute os procedimentos de forma a estudar o efeito da sobreutilização de índices.
 - a. Quanto demora a primeira inserção dos dados da tabela (sem índices)? E a segunda inserção de dados (com índices)? Porque que com índices demora mais tempo?
 - b. Se tivesse que escolher entre colocar um índice na coluna *LastName* ou na coluna *phone*, qual escolheria e porquê?
2. Abra o script “BTree.sql” e execute os procedimentos para estudar o exemplo de um B+ Tree no SQL Server.
 - a. Após a inserção de dois registos, verifique o estado do índice (ponto 1 do script “BTree.sql”). Insira 100 registos (ponto 2 do script “BTree.sql”), que tipo de índice existe? E com 700 registos (ponto 3 do script “BTree.sql”), mantêm-se o mesmo tipo de índice?

ETAPA 4

Para executar esta etapa ver anexo 1.

1. Ative o SQL Server Profiler e execute a seguinte consulta:

```
SELECT c.CompanyName  
FROM SalesLT.Customer c  
WHERE SalesPerson LIKE 'adventure-works\david%'
```

2. Verifique a informação disponível no SQL Server Profiler e grave o Profile para ser analisado no ponto 3.
3. Analise o trace no SQL Tuning Advisor. Faz sentido implementar o índice sugerido? Justifique.

(fim de enunciado)