Lab 3 - Índices e Monitorização

Nome: Nuno Reis Número: 202000753

Etapa 1

--1)

select object\_name(object\_id) as TableName, name As ColumnName, is\_identity

from sys.columns

where object\_name(object\_id)='Customer'

and name='CustomerID'

--2)

EXEC sys.sp\_helpindex @objname = N'SalesLT.Customer'

--3)--

SELECT 1.00 / (CAST(COUNT(DISTINCT CustomerID) AS DECIMAL(10,4))/ COUNT(CustomerID)) as CustomerID

FROM SalesLT.Customer;

SELECT 1.00 / (CAST(COUNT(DISTINCT NameStyle) AS DECIMAL(10,4))/ COUNT(NameStyle)) as NameStyle

FROM SalesLT.Customer;

SELECT 1.00 / (CAST(COUNT(DISTINCT Title) AS DECIMAL(10,4))/ COUNT(Title)) as Title

FROM SalesLT.Customer;

SELECT 1.00 / (CAST(COUNT(DISTINCT FirstName) AS DECIMAL(10,4))/ COUNT(FirstName)) as FirstName

FROM SalesLT.Customer;

SELECT 1.00 / (CAST(COUNT(DISTINCT MiddleName) AS DECIMAL(10,4))/ COUNT(MiddleName)) as MiddleName

FROM SalesLT.Customer;

SELECT 1.00 / (CAST(COUNT(DISTINCT LastName) AS DECIMAL(10,4))/ COUNT(LastName)) as LastName

FROM SalesLT.Customer;

SELECT 1.00 / (CAST(COUNT(DISTINCT Suffix) AS DECIMAL(10,4))/ COUNT(Suffix)) as Suffix

FROM SalesLT.Customer;

SELECT 1.00 / (CAST(COUNT(DISTINCT CompanyName) AS DECIMAL(10,4))/ COUNT(CompanyName)) as CompanyName

FROM SalesLT.Customer;

SELECT 1.00 / (CAST(COUNT(DISTINCT SalesPerson) AS DECIMAL(10,4))/ COUNT(SalesPerson)) as SalesPerson

FROM SalesLT.Customer;

SELECT 1.00 / (CAST(COUNT(DISTINCT EmailAddress) AS DECIMAL(10,4))/ COUNT(EmailAddress)) as EmailAddress

FROM SalesLT.Customer;

SELECT 1.00 / (CAST(COUNT(DISTINCT Phone) AS DECIMAL(10,4))/ COUNT(Phone)) as Phone

FROM SalesLT.Customer;

SELECT 1.00 / (CAST(COUNT(DISTINCT PasswordHash) AS DECIMAL(10,4))/ COUNT(PasswordHash)) as PasswordHash

FROM SalesLT.Customer;

SELECT 1.00 / (CAST(COUNT(DISTINCT PasswordSalt) AS DECIMAL(10,4))/ COUNT(PasswordSalt)) as PasswordSalt

FROM SalesLT.Customer;

SELECT 1.00 / (CAST(COUNT(DISTINCT rowguid) AS DECIMAL(10,4))/ COUNT(rowguid)) as rowguid

FROM SalesLT.Customer;

SELECT 1.00 / (CAST(COUNT(DISTINCT ModifiedDate) AS DECIMAL(10,4))/ COUNT(ModifiedDate)) as ModifiedDate

FROM SalesLT.Customer;

--4)

--CustomerID, FirstName, LastName, Suffix, CompanyName, EmailAddress, Phone, PasswordSalt, rowguid

Etapa 2

--1)

SET STATISTICS IO ON

SELECT c.LastName , c.FirstName

FROM SalesLT.Customer c

WHERE CustomerID=100

--King King

SELECT c.LastName , c.FirstName

FROM SalesLT.Customer c

WHERE c.phone ='979-555-0163'

--King King

--King King

--utilizando indices foram feitas 2 leituras enquanto que sem a utilização de indices foram feitas 60

--a utilização dos indices foi mais rapido e mais eficiente

--2)

IF EXISTS (SELECT name FROM sys.indexes

WHERE name = N'NONCI\_phone')

DROP INDEX NONCI\_phone ON SalesLT.Customer;

GO

CREATE NONCLUSTERED INDEX NONCI\_phone

ON SalesLT.Customer (phone);

GO

SELECT c.LastName , c.FirstName

FROM SalesLT.Customer c WITH(INDEX(NONCI\_phone))

WHERE CustomerID=100

SELECT c.LastName , c.FirstName

FROM SalesLT.Customer c WITH(INDEX(NONCI\_phone))

WHERE c.phone ='979-555-0163'

--

--3)

SELECT c.LastName , c.FirstName, c.Phone

FROM SalesLT.Customer c

WHERE c.phone like '96%'

Uma imagem com texto

Descrição gerada automaticamente

--4)

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

CREATE NONCLUSTERED INDEX NONCI\_lastName

ON SalesLT.Customer (LastName);

GO

--Ambas as querys usam o LastName ou para ordenar ou para selecionar

Etapa 3

--1)

/\*a)

Sem indices

SQL Server parse and compile time:

CPU time = 16 ms, elapsed time = 42 ms.

SQL Server Execution Times:

CPU time = 0 ms, elapsed time = 14 ms.

Com indices

SQL Server parse and compile time:

CPU time = 3 ms, elapsed time = 3 ms.

SQL Server Execution Times:

CPU time = 63 ms, elapsed time = 169 ms.

Ao usar o Execution plan foi facil ver que sem os indices os dados foram inseridos diretamente na tabela

enquanto que com os indices foi necessario separar e ordenar os dados por cada indice antes de serem inseridos

\*/

/\*b)

Colocava na coluna Phone porque é necessario ordenar menos dados logo é mais rapida e eficiente

\*/

--2)

Etapa 4

--1)

SELECT c.CompanyName

FROM SalesLT.Customer c

WHERE SalesPerson LIKE 'adventure-works\david%'

--2)

--3)

-- Faz sentido implementar o índice sugerido