**פרק 19 – Indexes and Statistics**

--1.

create clustered index idx\_MemberSessions

on Operation.MemberSessions (MemberId, LoginDateTime)

with (FILLFACTOR = 80) ON [PRIMARY]

--2.

select \*

from [Operation].[MemberSessions]

where MemberId = 1234

--3.

select \*

from [Operation].[MemberSessions] ms inner join [Operation].[Members] m

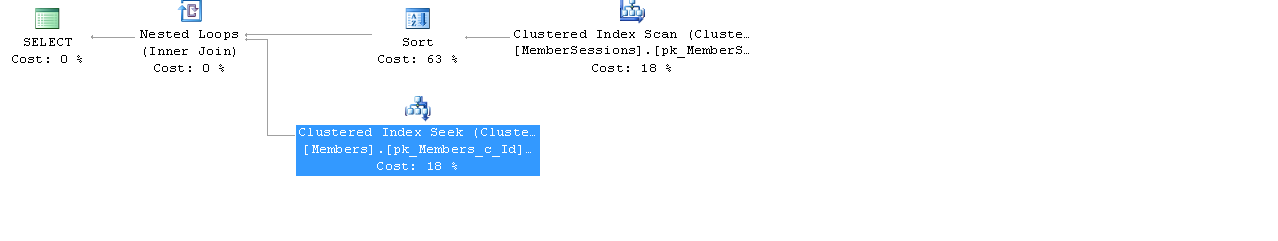
on ms.MemberId = m.Id

where m.GenderId = 2

and m.MaritalStatusId = 1

and datediff(yy,m.birthdate,getdate()) = 25

order by ms.MemberId, ms.LoginDateTime



--4.

create unique INDEX [ix\_Members\_UsernamePassword]

ON [Operation].[Members] (username, password)

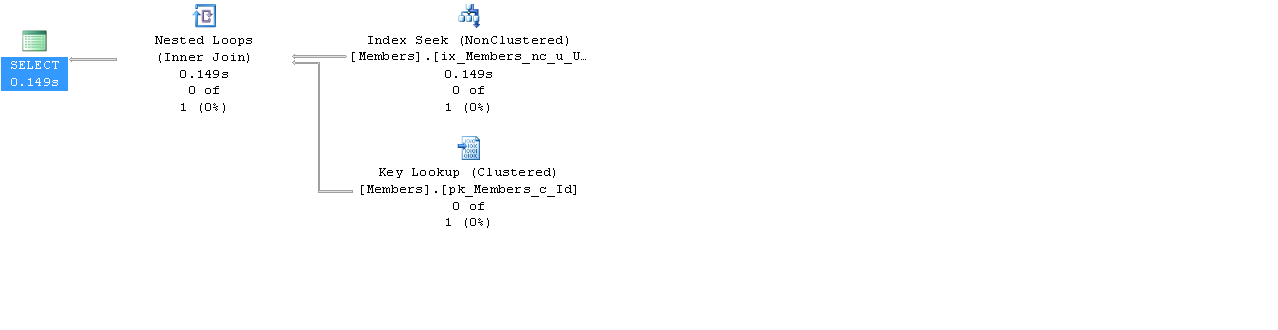
--5

select \*

from [Operation].[Members]

where Username = 'XXXX'

and password = '12345'



--6. --Cannot insert duplicate key

insert into [Operation].[Members]

values ('XXXXXXX',

'2144299',

'ggg','gfd',null,1,1234214,'xxxxxxx@gmail.com', 2,

GETDATE(),2,null,null,GETDATE())

--7

DBCC SHOW\_STATISTICS ('[Operation].[Members]', 'countryid')

--8

select i.Id

from Operation.Invitations i inner join Operation.Members m

on i.ReceivingMemberId = m.Id

where m.FirstName= 'Paul'

and m.LastName = 'Simon'

CREATE NONCLUSTERED INDEX [ix\_FirstLast\_Members]

ON [Operation].[Members](firstName, LastName)

--9

select \*

from Operation.Members

where id in (5,6,7,8)

create unique clustered INDEX [ix\_filterId\_Members]

ON [Operation].[Members] (id)

--10

select FirstName, LastName

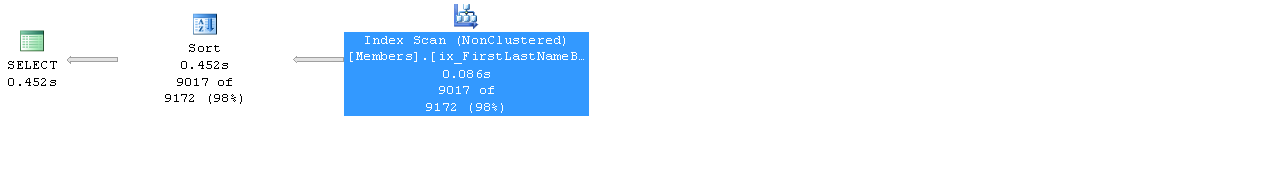
from Operation.Members

where LastName like 'B%'

order by FirstName , BirthDate desc,LastName

create INDEX [ix\_FirstLastNameBirth\_Members]

ON [Operation].[Members] (FirstName asc, BirthDate, LastName asc)



--11

select distinct FirstName, LastName

from Operation.Members

where LastName like '%b'

order by LastName, FirstName

--12

select \*,

substring(EmailAddress ,1, isGmails-1)

from Operation.Members

where EmailAddress not like '%@gmail.com'

drop INDEX [ix\_FirstLast\_Members]

ON [Operation].[Members]

CREATE NONCLUSTERED INDEX [ix\_EmailAddress\_Members]

ON [Operation].[Members](EmailAddress)

where EmailAddress <> '%@gmail.com'

--13

select BirthDate

from Operation.Members

where datediff(yy,birthdate,getdate()) > 50

and StreetAddress is null

drop INDEX ix\_Homeless\_Members on [Operation].[Members]

CREATE NONCLUSTERED INDEX [ix\_Homeless\_Members]

ON [Operation].[Members](StreetAddress)

INCLUDE (birthdate)

WHERE StreetAddress IS NULL

--14

drop index [ix\_filterId\_Members] ON [Operation].[Members]

create clustered index idx\_idFillFact\_Members

on Operation.Members (id) with (FILLFACTOR = 100)

ON [PRIMARY]

--15

create nonclustered index idx\_id\_Members

on [Operation].[MemberSessions] (EndDateTime asc, EndReasonId asc)

include (MemberId) with (FILLFACTOR = 80)

ON [PRIMARY]

--16

Select memberid = members.Id,

memberfistname = members.FirstName,

memberlastname = members.LastName,

logindatetime = MemberSessions.LoginDateTime,

sessionenddate = MemberSessions.EndDateTime

from Operation.Members as members

inner join

Operation.MemberSessions as MemberSessions

on

members.Id = MemberSessions.MemberId

where

members.CountryId = 4

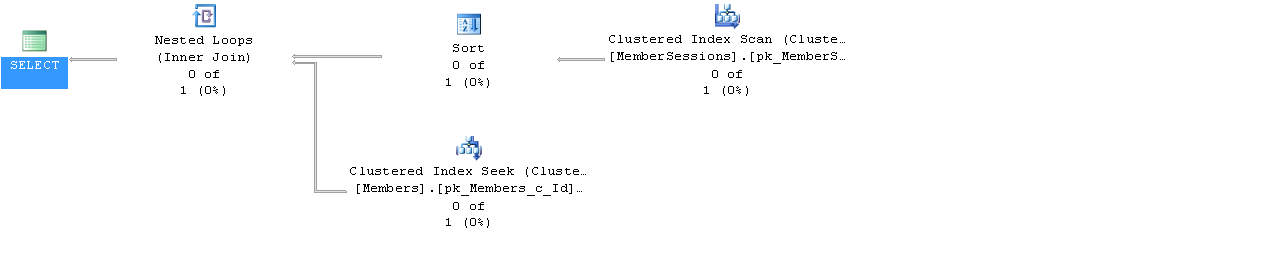
and

year(membersessions.logindatetime) = 2010

and

month(membersessions.logindatetime) = 6

order by memberid asc, LoginDateTime asc



--17

create nonclustered index idx\_loginEndDate\_MemberSessions

on [Operation].[MemberSessions] (MemberId, LoginDateTime asc, EndDateTime asc)

with (FILLFACTOR = 80)

ON [PRIMARY]

create nonclustered index idx\_IdCountryId\_Members

on [Operation].[Members] (id asc, CountryId) with (FILLFACTOR = 80)

ON [PRIMARY]

--18

SELECT OBJECT\_NAME(S.object\_id) AS table\_name, I.name AS index\_name,

i.type\_desc AS index\_type, IIF(i.is\_unique=1,'unique','notUnique'), i.fill\_factor

FROM sys.dm\_db\_index\_usage\_stats AS S

INNER JOIN sys.indexes AS i

ON S.object\_id = I.object\_id

AND S.index\_id = I.index\_id

--19

DBCC SHOW\_STATISTICS ('[Operation].[Members]', 'countryid')

--20

create table checkfillfactor

(

filler1 varCHAR(100) NOT NULL

);

GO

DECLARE @i AS int = 0;

WHILE @i < 1000

BEGIN

SET @i = @i + 1;

INSERT INTO checkfillfactor

(filler1)

VALUES

(@i);

END;

create nonclustered index idx\_filler1\_checkfillfactor

on dbo.checkfillfactor (filler1) with (FILLFACTOR = 80)

ON [PRIMARY]

create nonclustered index idx\_filler1\_2\_checkfillfactor

on dbo.checkfillfactor (filler1) with (FILLFACTOR = 10)

ON [PRIMARY]

select i.name, i.fill\_factor, OBJECT\_NAME(s.object\_id) as tableName,s.reserved\_page\_count, s.used\_page\_count, s.row\_count

from sys.dm\_db\_partition\_stats s inner join sys.indexes i

on OBJECT\_NAME(s.object\_id) = OBJECT\_NAME(i.object\_id)

where OBJECT\_NAME(s.object\_id)='checkfillfactor'

and i.type\_desc <> 'heap'

and s.index\_id > 1

--21

select avg\_fragmentation\_in\_percent as frag\_percent, avg\_fragment\_size\_in\_pages as frag\_size,

page\_count as index\_size

from sys.dm\_db\_index\_physical\_stats(db\_id('edate'),

object\_id('checkfillfactor'),null,null,null)

--22

SELECT db\_name(t.schema\_id) as [schema\_name], OBJECT\_NAME(S.object\_id) AS table\_name, I.name AS index\_name,

i.type\_desc AS index\_type, IIF(i.is\_unique=1,'unique','notUnique') as [unique/not unique], i.fill\_factor

FROM sys.dm\_db\_index\_usage\_stats AS S

INNER JOIN sys.indexes AS i

ON S.object\_id = I.object\_id

inner join sys.tables t

on i.object\_id = t.object\_id

AND S.index\_id = I.index\_id

--23

drop table A

create table A

(

filler1 CHAR(10) NOT NULL

);

GO

DECLARE @i AS int = 0;

WHILE @i < 10000

BEGIN

SET @i = @i + 1;

INSERT INTO A

(filler1)

VALUES

(FORMAT(@i,'0000'));

END;

CREATE TABLE [dbo].[A](

[A] [char](10) NOT NULL,

[Computed] AS ([A]+[A]) PERSISTED NOT NULL

) ON [PRIMARY]

GO

--show table size

SELECT

s.Name AS SchemaName,

t.NAME AS TableName,

p.rows AS RowCounts,

SUM(a.total\_pages) \* 8 AS TotalSpaceKB,

SUM(a.used\_pages) \* 8 AS UsedSpaceKB,

(SUM(a.total\_pages) - SUM(a.used\_pages)) \* 8 AS UnusedSpaceKB

FROM

sys.tables t

INNER JOIN

sys.schemas s ON s.schema\_id = t.schema\_id

INNER JOIN

sys.indexes i ON t.OBJECT\_ID = i.object\_id

INNER JOIN

sys.partitions p ON i.object\_id = p.OBJECT\_ID AND i.index\_id = p.index\_id

INNER JOIN

sys.allocation\_units a ON p.partition\_id = a.container\_id

WHERE

t.NAME = 'A'

GROUP BY

t.Name, s.Name, p.Rows

ORDER BY

s.Name, t.Name

--24.

a. yes- you can birthdate not null in filtered index

b. no- like is not allowed in filtered index

--25

SELECT count(\*) as [Indexes],

(select count(\*) from sys.indexes where has\_filter = 1) as [Filtered],

(select count(\*) FROM sys.indexes where type\_desc = 'CLUSTERED') as [Clustered],

(select count(\*) FROM sys.indexes where type\_desc = 'NONCLUSTERED') as [NonClustered],

(select count(\*) FROM sys.indexes where is\_unique=1) as [Unique],

(select count(\*) FROM sys.indexes where is\_unique=0) as [NotUnique],

(select count(\*) FROM sys.indexes where is\_primary\_key=1) as [PK],

(select count(\*) FROM sys.indexes where is\_primary\_key=0) as [NotPK]

FROM sys.indexes

where type\_desc <> 'heap'

--26

select index\_id, (select count(\*) from sys.dm\_db\_index\_physical\_stats (DB\_ID(), null, null, null, 'LIMITED') where avg\_fragmentation\_in\_percent >10)

from sys.dm\_db\_index\_physical\_stats (DB\_ID(), null, null, null, 'LIMITED')

--28.

SELECT i.name as indexName, OBJECT\_NAME(i.OBJECT\_ID) as tableName, i.type\_desc as IndexType,

(select count(\*) FROM sys.indexes where is\_unique=1) as [UniqueTable],

(select count(\*) FROM sys.indexes where is\_unique=0) as [NotUniqueTable],

s.system\_scans as noOfScans, o.create\_date as dateOfCreation

FROM sys.indexes as i

inner join sys.dm\_db\_index\_usage\_stats as s

on S.object\_id = I.object\_id

inner join sys.objects AS O

on O.object\_id = i.object\_id

where i.name <> 'null'

group by i.name, i.object\_id, i.index\_id, s.system\_scans, o.create\_date, i.type\_desc