/\*\*\*\*\*\* Script for SelectTopNRows command from SSMS \*\*\*\*\*\*/

SELECT TOP 1000 [SlNo]

,[ServerName]

,[SQLInstanceName]

,[DOMAIN]

,[PortNumber]

,[ServicePack]

,[VersionNumber]

,[Edition]

,[Environment]

,[Patch\_Cycle]

,[i\_Qualified]

,[i\_Active]

,[SQLserverVersion]

,[ApplicationOwner]

,[ApplicationContacts]

,[Comment]

,[Deviations]

,[rowguid]

FROM [ZBTAdmin].[dbo].[ServerDetails]

where [SQLserverVersion]='SQL Server 2008R2'

use master

go

set nocount on;

declare @dbfilename varchar(255), @dbfilesize int, @dbspacefree int, @dbpercentfree decimal(18,2)

create table #tmpcommand (commandname varchar(2048))

go

create table #tmpspaceused (dbname nvarchar(255), filenme varchar(255), spaceused float)

create table #db (name nvarchar(255))

declare @dbname sysname, @rc int

declare @cmdsql varchar(2000)

insert into #db (name) select name from master.sys.databases where

name not in ('master', 'model', 'msdb', 'tempdb') and

name not like '%reportserver%' and

databasepropertyex([name], 'isinstandby') = 0 and

databasepropertyex([name], 'status') = 'online'

select @rc = 1, @dbname = min(name) from #db

while @rc <> 0

begin

set @cmdsql = 'use [' + @dbname + '];' + 'insert into #tmpspaceused (dbname, filenme, spaceused) select ''' + @dbname + ''', name, fileproperty(name, ''spaceused'') from [' + @dbname + '].sys.sysfiles'

exec(@cmdsql)

--select \* from #tmpspaceused

select top 1 @dbname = name from #db where name > @dbname order by name

set @rc = @@rowcount

end

drop table #db

insert into #tmpcommand(commandname)

select 'use [' + a.name + '];' + ' dbcc shrinkfile(' +

b.name + ', ' + cast( ( cast((b.size \* 8 / 1024.0) as decimal(18,0)) - cast((b.size \* 8 / 1024.0) - (d.spaceused / 128.0) as decimal(15,0)) + 1 ) as varchar(20) )+ ')'

from sys.databases a

join sys.master\_files b on a.database\_id = b.database\_id

join #tmpspaceused d on a.name = d.dbname and b.name = d.filenme

--where cast((b.size \* 8 / 1024.0) - (d.spaceused / 128.0) as decimal(15,2)) / cast(cast((b.size \* 8 / 1024.0) as decimal(18,2)) as varchar(20)) > 0.01

and b.type\_desc ='rows'

drop table #tmpspaceused

declare @printcommand varchar(8000)

declare print\_cursor cursor for

select commandname from #tmpcommand order by commandname

open print\_cursor

fetch next from print\_cursor into @printcommand

while (@@fetch\_status <> -1)

begin

if (@@fetch\_status <> -2)

begin

print @printcommand

--exec @printcommand

end

fetch next from print\_cursor into @printcommand

end

drop table #tmpcommand

close print\_cursor

deallocate print\_cursor

declare @DefaultData nvarchar(512)

exec master.dbo.xp\_instance\_regread N'HKEY\_LOCAL\_MACHINE', N'Software\Microsoft\MSSQLServer\MSSQLServer', N'DefaultData', @DefaultData output

declare @DefaultLog nvarchar(512)

exec master.dbo.xp\_instance\_regread N'HKEY\_LOCAL\_MACHINE', N'Software\Microsoft\MSSQLServer\MSSQLServer', N'DefaultLog', @DefaultLog output

declare @DefaultBackup nvarchar(512)

exec master.dbo.xp\_instance\_regread N'HKEY\_LOCAL\_MACHINE', N'Software\Microsoft\MSSQLServer\MSSQLServer', N'BackupDirectory', @DefaultBackup output

declare @MasterData nvarchar(512)

exec master.dbo.xp\_instance\_regread N'HKEY\_LOCAL\_MACHINE', N'Software\Microsoft\MSSQLServer\MSSQLServer\Parameters', N'SqlArg0', @MasterData output

select @MasterData=substring(@MasterData, 3, 255)

select @MasterData=substring(@MasterData, 1, len(@MasterData) - charindex('\', reverse(@MasterData)))

declare @MasterLog nvarchar(512)

exec master.dbo.xp\_instance\_regread N'HKEY\_LOCAL\_MACHINE', N'Software\Microsoft\MSSQLServer\MSSQLServer\Parameters', N'SqlArg2', @MasterLog output

select @MasterLog=substring(@MasterLog, 3, 255)

select @MasterLog=substring(@MasterLog, 1, len(@MasterLog) - charindex('\', reverse(@MasterLog)))

select

isnull(@DefaultData, @MasterData) DefaultData,

isnull(@DefaultLog, @MasterLog) DefaultLog,

isnull(@DefaultBackup, @MasterLog) DefaultBackup

USE Final\_Audit\_data

GO

SELECT db.[name] AS [Final\_Audit\_data]

,id.[object\_id] AS [ObjectID]

,OBJECT\_NAME(id.[object\_id], db.[database\_id]) AS [ObjectName]

,id.[statement] AS [FullyQualifiedObjectName]

,id.[equality\_columns] AS [EqualityColumns]

,id.[inequality\_columns] AS [InEqualityColumns]

,id.[included\_columns] AS [IncludedColumns]

,gs.[unique\_compiles] AS [UniqueCompiles]

,gs.[user\_seeks] AS [UserSeeks]

,gs.[user\_scans] AS [UserScans]

,gs.[last\_user\_seek] AS [LastUserSeekTime]

,gs.[last\_user\_scan] AS [LastUserScanTime]

,gs.[avg\_total\_user\_cost] AS [AvgTotalUserCost] -- Average cost of the user queries that could be reduced by the index in the group.

,gs.[avg\_user\_impact] AS [AvgUserImpact] -- The value means that the query cost would on average drop by this percentage if this missing index group was implemented.

,gs.[system\_seeks] AS [SystemSeeks]

,gs.[system\_scans] AS [SystemScans]

,gs.[last\_system\_seek] AS [LastSystemSeekTime]

,gs.[last\_system\_scan] AS [LastSystemScanTime]

,gs.[avg\_total\_system\_cost] AS [AvgTotalSystemCost]

,gs.[avg\_system\_impact] AS [AvgSystemImpact] -- Average percentage benefit that system queries could experience if this missing index group was implemented.

,gs.[user\_seeks] \* gs.[avg\_total\_user\_cost] \* (gs.[avg\_user\_impact] \* 0.01) AS [IndexAdvantage]

,'CREATE INDEX [IX\_' + OBJECT\_NAME(id.[object\_id], db.[database\_id]) + '\_' + REPLACE(REPLACE(REPLACE(ISNULL(id.[equality\_columns], ''), ', ', '\_'), '[', ''), ']', '') + CASE

WHEN id.[equality\_columns] IS NOT NULL

AND id.[inequality\_columns] IS NOT NULL

THEN '\_'

ELSE ''

END + REPLACE(REPLACE(REPLACE(ISNULL(id.[inequality\_columns], ''), ', ', '\_'), '[', ''), ']', '') + '\_' + LEFT(CAST(NEWID() AS [nvarchar](64)), 5) + ']' + ' ON ' + id.[statement] + ' (' + ISNULL(id.[equality\_columns], '') + CASE

WHEN id.[equality\_columns] IS NOT NULL

AND id.[inequality\_columns] IS NOT NULL

THEN ','

ELSE ''

END + ISNULL(id.[inequality\_columns], '') + ')' + ISNULL(' INCLUDE (' + id.[included\_columns] + ')', '') AS [ProposedIndex]

,CAST(CURRENT\_TIMESTAMP AS [smalldatetime]) AS [CollectionDate]

FROM [sys].[dm\_db\_missing\_index\_group\_stats] gs WITH (NOLOCK)

INNER JOIN [sys].[dm\_db\_missing\_index\_groups] ig WITH (NOLOCK) ON gs.[group\_handle] = ig.[index\_group\_handle]

INNER JOIN [sys].[dm\_db\_missing\_index\_details] id WITH (NOLOCK) ON ig.[index\_handle] = id.[index\_handle]

INNER JOIN [sys].[databases] db WITH (NOLOCK) ON db.[database\_id] = id.[database\_id]

WHERE db.[database\_id] = DB\_ID()

--AND OBJECT\_NAME(id.[object\_id], db.[database\_id]) = 'YourTableName'

ORDER BY ObjectName, [IndexAdvantage] DESC

OPTION (RECOMPILE)