O post de hoje

Como de costume selecionei os principais scripts armazenados recentemente na minha biblioteca de códigos, que apresentam os seguintes assuntos:

Cálculo de Idade;

Cláusula Values;

Cláusula Where;

Comando Create Procedure;

Comando Declare;

Comando Exec;

Comando Group By;

Comando Insert;

Comando Order By;

Comando Pivot;

Comando Select;

Common Table Expression;

Diretiva Set;

DMV sys.dm\_os\_ring\_buffers;

DMV sys.dm\_os\_virtual\_address\_dump;

Filegroup;

Função Concat();

Função DateDiff();

Função de Agregação SUM;

Função Month();

Função Year();

Gerenciamento de Memória;

Grupo de Arquivos;

Inner Joins;

Left Joins;

Operador Lógico Condicional Case;

Operador Lógico Condicional While;

Stored Procedure;

Variáveis;

Variável do Tipo Tabela;

Visão;

Visão de Sistema sys.all\_objects;

Visão de Sistema sys.filegroups;

Visão de Sistema sys.indexes;

Visão de Sistema sys.syslanguages;

Visão de Sistema SYSCACHEOBJECTS; e

Visão de Sistemas sys.messages.

Chegou a hora, mãos nos teclados, a seguir apresento os códigos e exemplos selecionados para o Short Script – Março 2019 . Vale ressaltar que todos os scripts publicados nesta sessão foram devidamente testados, mas isso não significa que você pode fazer uso dos mesmo em seu ambiente de produção, vale sim todo cuidado possível para evitar maiores problemas.

Fique a vontade para compartilhar, comentar e melhorar cada um destes códigos.

Short Scripts

— Short Script 1 – Alterando a sequência numérica com base em uma condição —

-- Criando a Tabela Valores --

Create Table Valores

(Contador Int Primary Key Identity(1,1),

CodigoCaracter Char(4),

SequenciaDeControle Int)

Go

-- Inserindo os registros com CódigoCaracter 0001 --

Insert Into Valores Values ('0001',1),('0001',2),('0001',3),('0001',4)

Go

-- Inserindo os registros com CódigoCaracter 0002 --

Insert Into Valores Values ('0002',5),('0002',6),('0002',7),('0002',8)

Go

-- Inserindo os registros com CódigoCaracter 0003 --

Insert Into Valores Values ('0003',9),('0003',10)

Go

-- Consultando os registros inseridos --

Select Contador, CodigoCaracter, SequenciaDeControle from Valores

Go

-- Declarando um bloco de execução para alterar os registros de acordo com uma condição --

Declare @ContadorRegistros Int,

@ContadorSequenciaDeControle Int,

@ValorColunaCodigoCaracter Char(4)

Set @ContadorRegistros=1

Set @ContadorSequenciaDeControle=1

While @ContadorRegistros <= (Select Count(CodigoCaracter) from Valores)

Begin

Set @ValorColunaCodigoCaracter=(Select CodigoCaracter From Valores Where Contador = @ContadorRegistros)

If (@ValorColunaCodigoCaracter = (Select CodigoCaracter From Valores Where Contador = @ContadorRegistros) And @ContadorRegistros > 1)

Set @ContadorSequenciaDeControle=@ContadorSequenciaDeControle + 1

Else

Set @ContadorSequenciaDeControle=0

Update Valores

Set SequenciaDeControle = @ContadorSequenciaDeControle

Where Contador = @ContadorRegistros

Set @ContadorRegistros += 1

End

-- Consultando os registros após a alteração --

Select Contador, CodigoCaracter, SequenciaDeControle from Valores

Go

— Short Script 2 – Utilizando Extended Events - Ring Buffer através da sessão system\_health para obter detalhes sobre conexões abertas —

Declare @LanguageID int

Select @LanguageID = lcid

From sys.syslanguages

Where name = @@Language

WITH RingBufferXML

As (SELECT CAST(record as xml) AS RecordXML

FROM sys.dm\_os\_ring\_buffers

WHERE ring\_buffer\_type= 'RING\_BUFFER\_CONNECTIVITY'),

RingBufferConnectivity

As (SELECT x.y.value('(/Record/@id)[1]', 'int') AS [RecordID],

x.y.value('(/Record/ConnectivityTraceRecord/RecordType)[1]', 'varchar(max)') AS RecordType,

x.y.value('(/Record/ConnectivityTraceRecord/RecordTime)[1]', 'datetime') AS RecordTime,

x.y.value('(/Record/ConnectivityTraceRecord/SniConsumerError)[1]', 'int') AS Error,

x.y.value('(/Record/ConnectivityTraceRecord/State)[1]', 'int') AS State,

x.y.value('(/Record/ConnectivityTraceRecord/Spid)[1]', 'int') AS SPID,

x.y.value('(/Record/ConnectivityTraceRecord/RemoteHost)[1]', 'varchar(max)') AS RemoteHost,

x.y.value('(/Record/ConnectivityTraceRecord/RemotePort)[1]', 'varchar(max)') AS RemotePort,

x.y.value('(/Record/ConnectivityTraceRecord/LocalHost)[1]', 'varchar(max)') AS LocalHost,

x.y.value('(/Record/ConnectivityTraceRecord/LoginTimersInMilliseconds/TotalTime)[1]', 'int') AS TotalTime,

x.y.value('(/Record/ConnectivityTraceRecord/LoginTimersInMilliseconds/EnqueueTime)[1]', 'int') AS EnqueueTime,

x.y.value('(/Record/ConnectivityTraceRecord/LoginTimersInMilliseconds/NetWritesTime)[1]', 'int') AS NetWritesTime,

x.y.value('(/Record/ConnectivityTraceRecord/LoginTimersInMilliseconds/NetReadsTime)[1]', 'int') AS NetReadsTime,

x.y.value('(/Record/ConnectivityTraceRecord/LoginTimersInMilliseconds/Ssl/TotalTime)[1]', 'int') AS SslTotalTime,

x.y.value('(/Record/ConnectivityTraceRecord/LoginTimersInMilliseconds/Sspi/TotalTime)[1]', 'int') AS SspiTotalTime,

x.y.value('(/Record/ConnectivityTraceRecord/LoginTimersInMilliseconds/TriggerAndResGovTime)[1]', 'int') AS TriggerAndResGovTime

FROM RingBufferXML

CROSS APPLY RecordXML.nodes('//Record') AS x(y))

SELECT RBC.\*, m.text

FROM RingBufferConnectivity RBC

LEFT JOIN sys.messages M ON

RBC.Error = M.message\_id AND M.language\_id = @LanguageID

WHERE RBC.RecordType IN ('Error', 'LoginTimers')

ORDER BY RBC.RecordTime DESC

Go

— Short Script 3 – Stored Procedure - Realizando o versionamento de código fonte —

CREATE PROCEDURE dbo.GetOne

AS

SELECT 1

GO

CREATE PROCEDURE dbo.GetOne;2 --Versionamento, recurso antigo

AS

SELECT 2

GO

exec dbo.GetOne;2 -- Executando a procedure versão 2

go

— Short Script 4 – Gerenciamento de Memória — Como encontrar quem está usandoo espaço de endereço virtual no seu SQL Server —

SELECT convert(varchar,getdate(),120) as [Timestamp], max(region\_size\_in\_bytes)/1024 [Total max contiguous block size in KB]

from sys.dm\_os\_virtual\_address\_dump

where region\_state = 0x00010000 --- MEM\_FREE

Go

select SUM(virtual\_memory\_reserved\_kb)/1024 as virtual\_memory\_reserved\_mb

from sys.dm\_os\_memory\_clerks

where type not like '%bufferpool%'

Go

With VASummary(Size,Reserved,Free) AS

(SELECT

Size = VaDump.Size,

Reserved = SUM(CASE(CONVERT(INT, VaDump.Base)^0)

WHEN 0 THEN 0 ELSE 1 END),

Free = SUM(CASE(CONVERT(INT, VaDump.Base)^0)

WHEN 0 THEN 1 ELSE 0 END)

FROM

(

SELECT CONVERT(VARBINARY, SUM(region\_size\_in\_bytes))

AS Size, region\_allocation\_base\_address AS Base

FROM sys.dm\_os\_virtual\_address\_dump

WHERE region\_allocation\_base\_address <> 0x0

GROUP BY region\_allocation\_base\_address

UNION

SELECT CONVERT(VARBINARY, region\_size\_in\_bytes), region\_allocation\_base\_address

FROM sys.dm\_os\_virtual\_address\_dump

WHERE region\_allocation\_base\_address = 0x0

)

AS VaDump

GROUP BY Size)

SELECT SUM(CONVERT(BIGINT,Size)\*Free)/1024 AS [Total avail Mem, KB] ,CAST(MAX(Size) AS BIGINT)/1024 AS [Max free size, KB]

FROM VASummary

WHERE Free <> 0

Go

SELECT SUM(PAGESUSED)\*8/1024 'MB of MemToLeave memory consumed by procedures' FROM MASTER.DBO.SYSCACHEOBJECTS

WHERE PAGESUSED >1

Go

— Short Script 5 – Criando um Pivot em conjunto com Visão com Sumarização —

Create Table Fornos

(Id Int Primary Key,

Defeito Varchar(20) Not Null,

Forno Varchar(20) Not Null,

Equipe Varchar(20) Not Null)

Go

Insert Into Fornos

Values (1,'DEFEITO A','FORNO 3','AZUL'),

(2,'DEFEITO A','FORNO 2','VERDE'),

(3,'DEFEITO B','FORNO 1','AZUL'),

(4,'DEFEITO A','FORNO 1','PRETO'),

(5,'DEFEITO B','FORNO 2','VERDE'),

(6,'DEFEITO B','FORNO 2','AZUL'),

(7,'DEFEITO A','FORNO 1','PRETO'),

(8,'DEFEITO A','FORNO 2','AZUL')

Go

Select \* From Fornos

Go

-- Gerando o Pivot --

Select \* from

(Select CONCAT(Defeito,' - ', Equipe) As DefeitosPorEquipe,

CONCAT(Defeito,' - ', Equipe) As 'Defeitos Agrupados Por Equipes',

Equipe,

Forno As 'Fornos'

From Fornos) As F

Pivot (Count(DefeitosPorEquipe) For Equipe In ([Azul],[Preto],[Verde])) as Pvt

Go

-- Adicionando os Totais --

Select \* from

(

Select CONCAT(Defeito,' - ', Equipe) As DefeitosPorEquipe,

CONCAT(Defeito,' - ', Equipe) As 'Defeitos Agrupados Por Equipes',

Equipe,

Forno As 'Fornos'

From Fornos

) As F

Pivot (Count(DefeitosPorEquipe) For Equipe In ([Azul],[Preto],[Verde])) as Pvt

Union All

Select 'Totais....', '---->',

Sum(Azul) As SomaAzul,

Sum(Preto) As SomaPreto,

Sum(Verde) As SomaVerde

From

(

Select CONCAT(Defeito,' - ', Equipe) As DefeitosPorEquipe,

Equipe

From Fornos

) As F

Pivot (Count(DefeitosPorEquipe) For Equipe In ([Azul],[Preto],[Verde])) as Pvt

Go

— Short Script 6 – Identificando o filegroup vinculado para cada tabela de usuário —

SELECT o.[name],

o.[type],

i.[name],

i.[index\_id],

f.[name]

FROM sys.indexes i INNER JOIN sys.filegroups f

ON i.data\_space\_id = f.data\_space\_id

INNER JOIN sys.all\_objects o

ON i.[object\_id] = o.[object\_id]

WHERE i.data\_space\_id = f.data\_space\_id

AND o.type = 'U' -- User Created Tables

GO

— Short Script 7 – Realizando o cálculo da idade —

-- Exemplo 1 --

declare @Hoje date

set @Hoje= cast (current\_timestamp as date);

SELECT CPF, DATA\_NASC,

case when month(@Hoje) > month(DATA\_NASC)

then datediff (year, DATA\_NASC, @Hoje)

when month(@Hoje) = month(DATA\_NASC) and day(@Hoje) >= day(DATA\_NASC)

then datediff (year, DATA\_NASC, @Hoje)

else datediff (year, DATA\_NASC, @Hoje) -1

end as Idade

from tb\_idade\_dez

Go

-- Exemplo 2 --

SELECT CPF, DATA\_NASC, (datediff (month, DATA\_NASC, @Hoje) / 12) as Idade from tb\_idade\_dez

Go

-- Exemplo 3 --

SELECT CPF, DATA\_NASC, datediff(day,DATA\_NASC,getdate())/365.15 as idade from TB\_IDADE\_JANEIRO

Go

-- Exemplo 4 --

declare @DATAS table (DataNasc date);

INSERT into @DATAS values

('19880101'), ('19880111'), ('19880112'), ('19880113'),

('19880213'), ('19880313'), ('19880413'), ('19880513'),

('19880613'), ('19880713'), ('19880813'), ('19880913'),

('19881013'), ('19881113'), ('19881213')

declare @Hoje date

set @Hoje= '20190112'

SELECT DataNasc, @Hoje as Hoje,

datediff(day, DataNasc, @Hoje)/365.15 as idade,

case when month(@Hoje) > month(DataNasc) then datediff (year, DataNasc, @Hoje)

when month(@Hoje) = month(DataNasc) and day(@Hoje) >= day(DataNasc) then datediff (year, DataNasc, @Hoje)

else

datediff (year, DataNasc, @Hoje) -1

end as Anos

from @DATAS

Go

Muito bem, missão mais que cumprida! Uma nova relação de short scripts acaba de ser compartilhada, mesmo sendo denominados short entre aspas “pequenos”, posso garantir que todos estes exemplos são de grande importância, apresentam um valor e conhecimento do mais alto nível.

Chegamos ao final de mais um Short Scripts, espero que este material possa lhe ajudar, ilustrando o uso de alguns recursos e funcionalidades do Microsoft SQL Server.

Acredito que você tenha observado que estes códigos são conhecidos em meu blog, todos estão relacionados aos posts dedicados ao Microsoft SQL Server publicados no decorrer dos últimos anos.

Boa parte deste material é fruto de um trabalho dedicado exclusivamente a colaboração com a comunidade, visando sempre encontrar algo que possa ser a solução de um determinado problema, bem como, a demonstração de como se pode fazer uso de um determinado recurso.