<http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/>

« [SQL SERVER – Simple Example of Recursive CTE](http://blog.sqlauthority.com/2008/07/28/sql-server-simple-example-of-recursive-cte/)

[SQLAuthority News – Author BirthDay – SQL Server Birthday](http://blog.sqlauthority.com/2008/07/30/sqlauthority-news-author-birthday-sql-server-birthday/) »

SQL SERVER – SQL SERVER – Simple Example of Recursive CTE – Part 2 – MAXRECURSION – Prevent CTE Infinite Loop

July 29, 2008 by [pinaldave](http://blog.sqlauthority.com/author/pinaldave/)

Yesterday I wrote about SQL SERVER – SQL SERVER – Simple Example of Recursive CTE. I right away received email from regular reader John Mildred that if I can prevent infinite recursion of CTE.

Sure! recursion can be limited. Use the option of MAXRECURSION.

USE AdventureWorks  
GO  
WITH Emp\_CTE AS (  
SELECT EmployeeID, ContactID, LoginID, ManagerID, Title, BirthDate  
FROM HumanResources.Employee  
WHERE ManagerID IS NULL  
UNION ALL  
SELECT e.EmployeeID, e.ContactID, e.LoginID, e.ManagerID, e.Title,e.BirthDate  
FROM HumanResources.Employee e  
INNER JOIN Emp\_CTE ecte ON ecte.EmployeeID = e.ManagerID  
)  
SELECT \*  
FROM Emp\_CTE OPTION (MAXRECURSION 5)  
GO

Now if your CTE goes beyond 5th recursion it will throw an error and stop executing. If you put MAXRECURSION value too low it may be possible before your desire result is accomplished and will throw an error.

For example if you change MAXRECURSION to value 3. It will throw following error.

***Msg 530, Level 16, State 1, Line 1  
The statement terminated. The maximum recursion 3 has been exhausted before statement completion.***

In summary MAXRECUSION is good way to protect your CTE to go into infinite loop.

Reference : **Pinal Dave (**[**http://blog.SQLAuthority.com**](http://blog.sqlauthority.com/)**)**

[About these ads](http://en.wordpress.com/about-these-ads/)

**Share:**

* [More](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/)

**Like this:**

Posted in [Pinal Dave](http://blog.sqlauthority.com/category/pinal-dave/), [SQL](http://blog.sqlauthority.com/category/technology/sql/), [SQL Authority](http://blog.sqlauthority.com/category/technology/sql-authority/), [SQL Joins](http://blog.sqlauthority.com/category/sql-joins/), [SQL Query](http://blog.sqlauthority.com/category/technology/sql-query/), [SQL Scripts](http://blog.sqlauthority.com/category/sql-scripts/), [SQL Server](http://blog.sqlauthority.com/category/technology/sql-server/),[SQL Tips and Tricks](http://blog.sqlauthority.com/category/technology/sql-tips-and-tricks/), [T SQL](http://blog.sqlauthority.com/category/technology/t-sql/), [Technology](http://blog.sqlauthority.com/category/technology/) | Tagged [CTE](http://blog.sqlauthority.com/tag/cte/) | 28 Comments

28 Responses

1. on [July 31, 2008 at 11:26 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-41131) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=41131#respond)http://2.gravatar.com/avatar/ed7938c0c55a7ff953a6ffb21bdf502c?s=48&d=identicon**Kristina**

Thanks for this and your previous examples regarding recursion on SQL server. I was fond of oracle’s connect by, and sad to see that it didn’t exist in SQL Server. This did the trick, though! Thanks again!

1. on [August 29, 2008 at 2:49 am](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-42141) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=42141#respond)http://1.gravatar.com/avatar/4437c6eef93267d3c1debabed845777a?s=48&d=identicon**Haridas**

It’s easy to parse delimited string with WITH statement

DECLARE @txt as nvarchar(100)  
SET @txt = ‘Banana;Chocolote;Milk;Bread’

;WITH ParseText( Ltxt, Rtxt) AS (

SELECT –anchor member  
Ltxt = LEFT (@txt, CHARINDEX( ‘;’, @txt) )  
,Rtxt = RIGHT(@txt, LEN(@txt) – CHARINDEX( ‘;’,@txt) )

UNION ALL –Recursive member  
SELECT

Ltxt = LEFT(tn.Rtxt, CHARINDEX( ‘;’,tn.Rtxt) )  
,Rtxt = RIGHT(tn.Rtxt, LEN(tn.Rtxt) – CHARINDEX( ‘;’,tn.Rtxt) )  
FROM ParseText tn where Ltxt ” — Recursive loop  
)  
SELECT  
Ltxt = CASE WHEN Ltxt = ” THEN RTxt ELSE Ltxt END  
FROM ParseText

* + on [March 14, 2010 at 1:41 am](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-62887) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=62887#respond)http://0.gravatar.com/avatar/c324a610e4bdd61b7cc03ba8f995f2a2?s=48&d=identicon**katie**

I tried this example, but it didn’t go into the recursive statements. I modified it to parse the entire string and return each element of the string.

DECLARE @txt as nvarchar(100)  
SET @txt = ‘Banana;Chocolate;Milk;Bread;’;

WITH ParseText( Ltxt, Rtxt) AS (

SELECT  
Ltxt = LEFT (@txt, CHARINDEX( ‘;’, @txt)-1 )  
,Rtxt = RIGHT(@txt, LEN(@txt) – CHARINDEX( ‘;’,@txt) )

UNION ALL  
SELECT  
Ltxt = LEFT(tn.Rtxt, CHARINDEX( ‘;’,tn.Rtxt)-1 )  
,Rtxt = RIGHT(tn.Rtxt, LEN(tn.Rtxt) – CHARINDEX( ‘;’,tn.Rtxt) )  
FROM ParseText tn  
where Rtxt ”  
)  
SELECT  
Ltxt = CASE WHEN Ltxt = ” THEN RTxt ELSE Ltxt END  
FROM ParseText

* + - on [March 14, 2010 at 1:43 am](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-62888) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=62888#respond)http://0.gravatar.com/avatar/c324a610e4bdd61b7cc03ba8f995f2a2?s=48&d=identicon**katie**

Btw, you need to replace the slanted single quotes to match the straight single quotes if you try to copy and paste the query into SSMS.

1. on [March 31, 2009 at 7:48 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-50291) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=50291#respond)http://0.gravatar.com/avatar/f3822cb358c8e3cf61575da6584466ef?s=48&d=identicon**Prasant**

Hi Pinal,

Thanks for the post on CTE. Still I am not clear how can we restrict the recursion in CTE. Ex. The normal select statement (without MAXRECURSION) returns 10 records. If we give OPTION (MAXRECURSION 3), the query will throw error. If we give MAXRECURSION 4 or above it gives all the 10 records which is same as query without MAXRECURSION. Could you please clarify how can it restrict the recursion to say 3?

1. on [April 1, 2009 at 1:28 am](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-50305) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=50305#respond)http://0.gravatar.com/avatar/6950dd8f71086248d263a44bec6e11a0?s=48&d=identicon**Brian Tkatch**

@Prasant

Please provide an example query.

1. on [August 14, 2009 at 12:27 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-54846) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=54846#respond)http://0.gravatar.com/avatar/cf1ff9c3a2d201bd1814d750c01ef52e?s=48&d=identicon[**Ravi kiran**](http://dot-guru.blogspot.com/)

Hi pinal,

Thank you for the content, i was looking for the same.

1. on [October 27, 2009 at 8:11 am](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-57055) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=57055#respond)http://0.gravatar.com/avatar/9a2eea17ee4ab2bdc36ba004bd1c84fd?s=48&d=identicon**Raghuveer**

Hi Pinal,

I am student learning SQL server. I actaully didnot know what exactly maxrecursion function do?

can you please be precise.

Thanks&Regards,

Raghuveer.

1. on [October 27, 2009 at 6:51 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-57071) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=57071#respond)http://0.gravatar.com/avatar/6950dd8f71086248d263a44bec6e11a0?s=48&d=identicon[**Brian Tkatch**](http://tkatch.com/)

@Raghuveer

A recursive CTE can potentially go on forever. So, SQL Server limits it to 100 cycles. MAXRECURSION allows you to change that limit between 1 and 32767 or 0 which means unlimited.

1. on [November 12, 2009 at 10:09 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-57557) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=57557#respond)http://1.gravatar.com/avatar/d7140124547d74d56a46de59c4689725?s=48&d=identicon**Alain Holloway**

Hi there!

I need help with recursive operation

I do have 2 tables

#1 Inspections  
ID  
PARENT\_ID  
POSITION

#2 Resources  
RESOURCES\_ID  
CULTURE\_LANGUAGE  
CAPTION

DATA SAMPLE (OUTPUT)

SELECT a.id, a.parent\_id, a.position, b.culture\_language, b.caption  
FROM Inspections AS a  
INNER JOIN InspectionsResources AS b ON  
a.id = b.resource\_id where culture\_language=’en’

VEHICLES ID = 1, PARENT\_ID = 0, POSITION=1  
FORKLIFTS ID = 2, PARENT\_ID = 0, POSITION=2  
CARS ID = 3, PARENT\_ID = 1, POSITION=1  
MOTORCYCLES ID = 4, PARENT\_ID = 1, POSITION=2  
DOORS ID = 5, PARENT\_ID = 3, POSITION=1  
HANDLES ID = 6, PARENT\_ID = 3, POSITION=2  
…

I need to load all data for CARS only!!

Thank you Pinal

Alain

1. on [February 16, 2010 at 12:18 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-61220) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=61220#respond)http://1.gravatar.com/avatar/a9b7d7596b9e5f30eaa484eb33a7f7d9?s=48&d=identicon[**Paresh Prajapati**](http://paresh-sqldba.blogspot.com/)

Hi Pinal,

How can we avoid the error  
“Msg 530, Level 16, State 1, Line 1  
The statement terminated. The maximum recursion 3 has been exhausted before statement completion. ”

if cte goes above 100 levels, then what need to do?

* + on [June 18, 2012 at 4:31 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-302778) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=302778#respond)http://0.gravatar.com/avatar/030c17532b6719a7b3b41711591f30f7?s=48&d=identicon[**madhivanan**](http://www.beyondrelational.com/blogs/madhivanan)

You need to set it to 0

1. on [February 16, 2010 at 11:43 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-61288) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=61288#respond)http://2.gravatar.com/avatar/2445944c24187140a40615dfbbb588dc?s=48&d=identicon[**Pinal Dave**](http://blog.sqlauthority.com/)

Hi Paresh,

Set the MAXRECURSION hint of CTE query to any nonnegative value between 0 and 32767 as below:

SELECT EmployeeID, ManagerID, Title  
FROM cteEmployee  
OPTION (MAXRECURSION 200);

When 0 is specified, no limit is applied. If this option is not specified, the default limit is 100.

Regards,  
Pinal Dave

* + on [February 17, 2010 at 12:10 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-61311) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=61311#respond)http://1.gravatar.com/avatar/a9b7d7596b9e5f30eaa484eb33a7f7d9?s=48&d=identicon[**Paresh Prajapati**](http://paresh-sqldba.blogspot.com/)

Hello Pinal,

It is really working.

Thanks a lot.

1. on [March 12, 2010 at 11:24 am](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-62789) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=62789#respond)http://0.gravatar.com/avatar/600d47f343d83a031ccdd9b693cb2432?s=48&d=identicon[**Niranjan**](http://www.kensistech.com/)

Hello Pinal Dave,

I have a problem with MAXRECURSION hint of CTE query  
below  
declare @id int  
set @id=1;

with cte(member\_id,lft,rgt) as (  
select member\_id,0,0  
from [MultilevelChain].[dbo].[member\_info]  
where member\_id=@id

union all

select t.member\_id,1-t.position, t.position  
from [MultilevelChain].[dbo].[member\_info] t  
inner join cte c on c.member\_id = t.sponser\_id)  
select sum(lft) as LeftNode,  
sum(rgt) as RightNode

from cte  
option (maxrecursion 32767)

it gives error message like  
Msg 530, Level 16, State 1, Line 5  
The statement terminated. The maximum recursion 32767 has been exhausted before statement completion.

Here all below levels are searched but problem with top levelit has attach with 6 node under it. all levels are ok but not show id as 1 pls help me. i waitig to ur reply

Tanks regards  
Niranjan Singh Pune

1. on [April 1, 2010 at 8:03 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-64238) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=64238#respond)http://2.gravatar.com/avatar/2a033eb4d0659604ec0ed22ea63bbf79?s=48&d=identicon**Luke Williamson**

I’m creating a .net cms application. I want a user to be able to build a menu by select multiple parent menu items. Each menu item can have two levels of inheritance.

Using cte I can get the tree structure for the menu, however I want to be able to insert the tree structure with positioning. As users with also be able to change the positioning of the menu items.

As under the news sub menu World news should be before UK news.

Any help and guidance will be greatly appreciated.

Is this even possible using CTE?

DECLARE @menu TABLE  
(  
id int, parentId int, title varchar(128)  
)

INSERT @menu  
SELECT 1,NULL,’Home Page’ UNION ALL  
SELECT 2,1,’News’ UNION ALL  
SELECT 3,2,’World’ UNION ALL  
SELECT 4,2,’UK’ UNION ALL  
SELECT 5,2,’Business’ UNION ALL  
SELECT 6,1,’Sports’ UNION ALL  
SELECT 7,6, ‘Football’ UNION ALL  
SELECT 8,6,’Tennis’ UNION ALL  
SELECT 9,6, ‘Formula One’ UNION ALL  
SELECT 10,6, ‘Boxing’ UNION ALL  
SELECT 11,2, ‘Politics’ UNION ALL  
SELECT 12,6, ‘Cricket’

;WITH MenuTree AS  
(  
SELECT  
M.id,  
M.parentId,  
M.title,  
0 AS menuLevel,  
Cast(M.title + ‘\\’ AS varchar(max)) AS treePath,  
null AS position  
FROM  
@menu AS M  
WHERE  
M.id = 1

UNION ALL

SELECT  
M.id,  
M.parentId,  
M.title,  
menuLevel + 1,  
Cast(MenuTree.treePath + M.title + ‘\\’ AS varchar(max)) AS treePath,  
null AS position  
FROM @menu AS M  
INNER JOIN MenuTree ON M.ParentID = MenuTree.id  
)

SELECT \* FROM menuTree order by treePath

how do i get the positioning…

id – title – position  
1 Home Page – 1  
2 News – 1  
5 Business – 2  
11 Politics – 3  
4 UK – 4  
3 World – 5  
6 Sports – 2  
10 Boxing – 1  
12 Cricket – 2  
7 Football – 3  
9 Formula One – 4  
8 Tennis -5

Thanks for your time.

1. on [May 4, 2010 at 1:16 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-68797) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=68797#respond)http://1.gravatar.com/avatar/4d4a8592e6dee110b264f03295c21d23?s=48&d=identicon[**John Bailey**](http://www.baileysc.com/)

I was hoping to restrict the recursion without throwing an error. Is there a way to get it to just stop at lets say 5 levels deep without throwing an error?

* + on [May 4, 2010 at 11:02 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-69040) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=69040#respond)http://1.gravatar.com/avatar/42d1a314228728297e2db729d3dfb7cb?s=48&d=identicon[**Marko Parkkola**](http://weicco.spaces.live.com/)

Well, of course. Just keep a counter which tells how deep in the recursion you are and stop there. For example like the following (which is taken and modified from example in my blog).

Take notice that ‘lvl’ field there which tells how deep we are. In recursion part I check that lvl < 2 and recursion stops there if it ever goes so deep.

WITH  
cte(emp\_id, emp\_name, sup\_id, sup\_name, lvl) AS  
(  
SELECT e.emp\_id, e.emp\_name, s.emp\_id, s.emp\_name, 1  
FROM t\_employee e  
LEFT OUTER JOIN t\_employee s ON s.emp\_id = e.supervisor\_id  
),  
cte\_start(emp\_id, emp\_name, sup\_id, sup\_name, lvl) AS  
(  
SELECT e.emp\_id, e.emp\_name, s.emp\_id, s.emp\_name, 1  
FROM t\_employee e  
JOIN t\_employee s ON s.emp\_id = e.supervisor\_id  
LEFT OUTER JOIN t\_employee sub ON sub.supervisor\_id = e.emp\_id  
WHERE sub.emp\_id IS NULL  
),  
cte\_recur(emp\_id, emp\_name, sup\_id, sup\_name, lvl) AS  
(  
SELECT emp\_id, emp\_name, sup\_id, sup\_name, lvl  
FROM cte\_start  
UNION ALL  
SELECT c.emp\_id, c.emp\_name, c.sup\_id, c.sup\_name, r.lvl + 1  
FROM cte c  
JOIN cte\_recur r ON r.sup\_id = c.emp\_id  
where r.lvl < 2  
)  
SELECT DISTINCT emp\_name 'Employee name', sup\_name 'Supervisor name', lvl  
FROM cte\_recur

1. on [May 4, 2010 at 5:09 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-68872) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=68872#respond)http://2.gravatar.com/avatar/2a033eb4d0659604ec0ed22ea63bbf79?s=48&d=identicon**Luke**

I think you can set a where clause on the level.

UNION ALL

SELECT  
whatever…  
WHERE  
menulevel < 5

1. on [November 28, 2010 at 9:13 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-102648) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=102648#respond)http://0.gravatar.com/avatar/fbced6c8bf31080699d6893c326bb235?s=48&d=identicon**Brijesh Shah**

Hi there!

I need help with recursive operation like

Parents 1  
———Chield1  
———Chield2  
——Sub -Chield1  
——Sub -Chield2  
Parents 2  
Parents3  
———Chield1  
———Chield2

I want to display data exactly like above example with ‘-’ operator.  
Please provide an query.  
thanks & regards  
Brijesh Shah

1. on [May 10, 2011 at 7:47 am](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-133698) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=133698#respond)http://0.gravatar.com/blavatar/08e35387c05b61340e885b1763a69d9f?s=48[**SQL SERVER – Common Table Expression (CTE) and Few Observation Journey to SQLAuthority**](http://blog.sqlauthority.com/2011/05/10/sql-server-common-table-expression-cte-and-few-observation/)

[...] SQL SERVER – SQL SERVER – Simple Example of Recursive CTE – Part 2 – MAXRECURSION – Preven… [...]

1. on [November 19, 2011 at 11:37 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-199407) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=199407#respond)http://1.gravatar.com/avatar/148a0b2a8165c62f8e15cf28f9891154?s=48&d=identicon**Dabbas**

Hi pinaldave  
It’s great article and very helpful.  
I’ve question.  
I’ve sub group and trying to get all the parents so I tried this:  
==================================  
WITH Groups\_CTE AS (  
SELECT SubG.ParentID,SubG.Name FROM [Groups] AS SubG WHERE SubG.ID = 1  
UNION ALL  
SELECT ParentG.ID,ParentG.Name FROM Groups AS ParentG INNER JOIN Groups\_CTE ecte ON ecte.ParentID = ParentG.ID  
)

SELECT \* FROM Groups\_CTE  
==================================  
But it didn’t work, it thow exception says:  
“The statement terminated. The maximum recursion 100 has been exhausted before statement completion.”

What should I do to make it work properly ?

Thanx

1. on [January 27, 2012 at 7:01 am](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-244329) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=244329#respond)http://0.gravatar.com/blavatar/08e35387c05b61340e885b1763a69d9f?s=48[**SQL SERVER – Common Gotcha’s Associated with Common Table Expressions (CTE) – Quiz – Puzzle – 26 of 31 « SQL Server Journey with SQL Authority**](http://blog.sqlauthority.com/2012/01/27/sql-server-common-gotchas-associated-with-common-table-expressions-cte-quiz-puzzle-26-of-31/)

[...] Multiple CTE in One SELECT Statement Query Delete Duplicate Rows Simple Example of Recursive CTE SQL SERVER – Simple Example of Recursive CTE – Part 2 – MAXRECURSION – Prevent CTE Infinite … T-SQL Paging Query Technique Comparison (OVER and ROW\_NUMBER()) – CTE vs. Derived [...]

1. on [February 8, 2012 at 7:02 am](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-249752) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=249752#respond)http://0.gravatar.com/blavatar/08e35387c05b61340e885b1763a69d9f?s=48[**SQL SERVER – Convert Subquery to CTE – SQL in Sixty Seconds #001 – Video « SQL Server Journey with SQL Authority**](http://blog.sqlauthority.com/2012/02/08/sql-server-convert-subquery-to-cte-sql-in-sixty-seconds-001-video/)

[...] One SELECT Statement Query Common Table Expression (CTE) and Few Observation Delete Duplicate Rows Simple Example of Recursive CTE – Part 2 – MAXRECURSION – Prevent CTE Infinite Loop T-SQL Paging Query Technique Comparison (OVER and ROW\_NUMBER()) – CTE vs. Derived [...]

1. on [June 8, 2012 at 7:32 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-297521) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=297521#respond)http://0.gravatar.com/avatar/3be890435e6e1a132959ab8448045e84?s=48&d=identicon**Prakash**

It mentioned that “Now if your CTE goes beyond 5th recursion it will throw an error and stop executing.” can you please elaborate a bit as I have made max recursion 200 but didn’t get any error..thnx

1. on [June 19, 2012 at 10:51 am](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-303044) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=303044#respond)http://2.gravatar.com/avatar/5a2815843eeb0ae3a6fa089f5e0cb20c?s=48&d=identicon[**Govind**](http://govindbadkur.blogspot.in/)

@Prakash,

there is mention “OPTION (MAXRECURSION 5)” so it will throw error beyond 5th recursion, while you have made max recursion to 200, so it will throw error after 200th recursion..

1. on [December 13, 2012 at 3:18 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-392427) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=392427#respond)http://1.gravatar.com/avatar/71a8512a9bc37b6a9f22c0871b81deac?s=48&d=identicon**Vishal bhargava**

I have a CTE Table Category and Sub Category With ID,Parent ID relation.

I Wana to retrieve all Nested Sub Categories In Nested Ordered Manner.

Some one Help me soon The Result Shuld be in Following Manners.

Main Cat

– Sub A of Main

—-Child Sub A of Sub A of Main

—-Child Sub B of Sub A of Main

—-Child Sub C of Sub A of Main

– Sub B of Main

—-Child Sub A of Sub B of Main

—-Child Sub B of Sub B of Main

—-Child Sub C of Sub B of Main

– Sub C of Main

—-Child Sub A of Sub C of Main

—-Child Sub B of Sub C of Main

—-Child Sub C of Sub C of Main

1. on [January 11, 2013 at 11:10 pm](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/#comment-406558) | [Reply](http://blog.sqlauthority.com/2008/07/29/sql-server-sql-server-simple-example-of-recursive-cte-part-2-maxrecursion-prevent-cte-infinite-loop/?replytocom=406558#respond)http://2.gravatar.com/avatar/e586f43b3576add24733d74059b04ca8?s=48&d=identicon**Dilshod**

It is very good example for starters. Thank you for posting!