

How to change string data with a mass update?

Asked 15 years, 11 months ago Modified 7 years, 3 months ago

Viewed 602 times



1



I have a table (T1) in t-sql with a column (C1) that contains almost 30,000 rows of data. Each column contains values like MSA123, MSA245, MSA299, etc. I need to run an update script so the MSA part of the string changes to CMA. How can I do this?



sql-server-2005

t-sql



Share

Improve this question

Follow

asked Dec 29, 2008 at 16:06



Xaisoft

46.5k ● 87 ● 284 ● 443

6 Answers

Sorted by:

Highest score (default)



3

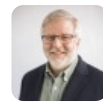
```
update t1
set c1 = replace(c1, "MSA", "CMA")
where c1 like "MSA%"
```



Share Improve this answer

answered Dec 29, 2008 at 16:10

Follow



tvanfosson

532k ● 102 ● 699 ● 798

oh sure.. spell it out for him! ;) – [Kon](#) Dec 29, 2008 at 16:10

who do you want me to give the answer to? – [Xaisoft](#) Dec 29, 2008 at 16:12

beat me to it, but I'll leave mine up for character range example. – [Joel Coehoorn](#) Dec 29, 2008 at 16:13

Your always very helpful Joel. – [Xaisoft](#) Dec 29, 2008 at 16:14



I don't have SQL Server in front of me, but I believe that this will work:

2



```
UPDATE T1 SET C1 = REPLACE(C1, 'MSA', 'CMA');
```



Share Improve this answer

edited Aug 31, 2017 at 20:45



Follow



C B

1,717 ● 6 ● 20 ● 20

answered Dec 29, 2008 at 16:15



Perry Neal

765 ● 4 ● 7



You can use [REPLACE](#) function to do it.

1

Share Improve this answer

answered Dec 29, 2008 at 16:09

Follow



Kon

27.4k ● 12 ● 63 ● 87



1

In addition to what fallen888 posted, if there are other values in that table/column as well you can use the LIKE operator in the where clause to make sure you only update the records you care about:



```
... WHERE [C1] LIKE 'MSA[0-9][0-9][0-9]'
```



Share Improve this answer

answered Dec 29, 2008 at 16:12

Follow



Joel Coehoorn

415k ● 114 ● 577 ● 813

Sometimes I wish T-SQL had better regexp semantics. It would be nice to be able to tie the MSA to the beginning of the string in the replace function, for instance. – [tvanfosson](#)
Dec 29, 2008 at 16:18

You could do that via an activeX or .Net object in a user defined function, but agree that it should be built-in.

– [Joel Coehoorn](#) Dec 29, 2008 at 16:33



0

While replace will appear to work, what happens when you need to replace M with C for MSA but not for MCA? Or if you have MSAD as well as MSA in the data right



now and you didn't want that changed (or CMSA). Do you even know for sure if you have any data being replaced that you didn't want replaced?



The proper answer is never to store data that way. First rule of database design is to only store one piece of information per field. You should have a related table instead. It will be easier to maintain over time.

Share Improve this answer

answered Dec 30, 2008 at 20:10

Follow



HLGEM

96.4k ● 15 ● 119 ● 189



0



I have to disagree with HLGEM's post. While it is true that the first normal form talks about atomicity in E.F.

Codd's original vision (which is the most controversial aspect of 1NF IMHO) the original request does not necessarily mean that there are no related tables or that the value is not atomic.



MSA123 may be the natural key of the object in question and the company may have simply decided to rename their product line. It is correct to say that if an artificial ID was used then even updates to the natural key would not require as many rows to be updated but if that is what you are implying then I would argue that artificial keys are definitely not the first rule of database design. They have their advantages but they also have many disadvantages which I won't go into here but a little googling would turn up quite a bit of controversy on whether or not to use artificial primary keys.

As for the original request, others have already nailed it,
REPLACE is the way to go.

Share Improve this answer

answered Dec 30, 2008 at 20:25

Follow



Shane Delmore

1,575 ● 2 ● 13 ● 19
