How to split a single column values to multiple column values?



I have a problem splitting single column values to multiple column values.

22

For Example:

Name



abcd efgh
ijk lmn opq
asd j. asdjja
asb (asdfas) asd

and I need the output something like this:

first_name	last_name
abcd	efgh
ijk	opq
asd	asdjja
asb	asd
asd	null

The middle name can be omitted (no need for a middle name) The columns are already created and need to insert the data from that single Name column.







Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Facebook

Need that in a select statement. Actually it is for stored procedure which is inserting the data by selecting values from a table. So I get it in select statement that will be great... - Shahsra Feb 25 '11 at 23:00

6 Answers



Your approach won't deal with lot of names correctly but...

19

10





```
SELECT CASE
     WHEN name LIKE '% %' THEN LEFT(name, Charindex(' ', name) - 1)
     ELSE name
END,
CASE
     WHEN name LIKE '% %' THEN RIGHT(name, Charindex(' ', Reverse(name)) - 1)
END
FROM YourTable
```

answered Feb 25 '11 at 23:05



I get the error : #1305 - FUNCTION preprodshem. Charindex does not exist. after some search I found that this function is not supported by mysql, and need some changes using : SUBSTRING_INDEX to resolve this. – Mimouni Sep 21 '16 at 10:09

2 @IlyasMimouni - this question is tagged SQL Server. – Martin Smith Sep 21 '16 at 10:41



An alternative to Martin's

13



Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up







```
create table somenames (Name varchar(100))
insert somenames select 'abcd efgh'
insert somenames select 'ijk lmn opq'
insert somenames select 'asd j. asdjja'
insert somenames select 'asb (asdfas) asd'
insert somenames select 'asd'
insert somenames select ''
insert somenames select null
```

edited Feb 25 '11 at 23:29

answered Feb 25 '11 at 23:10



So this guery works for millions of values in that column I guess right.... - Shahsra Feb 25 '11 at 23:16

@Shahsra - Yes. By definition SQL is a SET-based language so what you do on one record is done on all records equally.. - RichardTheKiwi Feb 25 '11 at 23:18

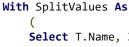
This appears to just give the forename twice at the moment? - Martin Smith Feb 25 '11 at 23:21

thank you very much Richard...Really appreciate it.... - Shahsra Feb 25 '11 at 23:22

@Martin - thanks for catching it. Fixed now. No longer short: (-RichardTheKiwi Feb 25 '11 at 23:30



What you need is a split user-defined function. With that, the solution looks like



```
Select T.Name, Z.Position, Z.Value
        , Row Number() Over ( Partition By T.Name Order By Z.Position ) As Num
   From Table As T
        Cross Apply dbo.udf Split( T.Name, ' ' ) As Z
Select Name
   , FirstName.Value
     Case When ThirdName Ts Null Then SecondName Flse ThirdName Fnd As LastName
```

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up or sign in with



Faceboo

```
And S2.Num = 3
Where FirstName.Num = 1
```

Here's a sample split function:

```
Create Function [dbo].[udf Split]
   @DelimitedList nvarchar(max)
    , @Delimiter nvarchar(2) = ','
RETURNS TABLE
AS
RETURN
   With CorrectedList As
        Select Case When Left(@DelimitedList, Len(@Delimiter)) <> @Delimiter Then
@Delimiter Else '' End
           + @DelimitedList
           + Case When Right(@DelimitedList, Len(@Delimiter)) <> @Delimiter Then
@Delimiter Else '' End
           As List
            , Len(@Delimiter) As DelimiterLen
         Numbers As
        Select TOP( Coalesce(DataLength(@DelimitedList)/2,0) ) Row Number() Over ( Order
By c1.object id ) As Value
        From sys.columns As c1
           Cross Join sys.columns As c2
   Select CharIndex(@Delimiter, CL.list, N.Value) + CL.DelimiterLen As Position
        , Substring (
                    CL.List
                    , CharIndex(@Delimiter, CL.list, N.Value) + CL.DelimiterLen
                    , CharIndex(@Delimiter, CL.list, N.Value + 1)
                        - ( CharIndex(@Delimiter, CL.list, N.Value) + CL.DelimiterLen )
                    ) As Value
   From CorrectedList As CL
        Cross Join Numbers As N
   Where N.Value <= DataLength(CL.List) / 2</pre>
```

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up or sign in with



Faceboo

answered Feb 25 '11 at 23:02





```
;WITH Split_Names (Name, xmlname)
AS
   SELECT
   Name,
   CONVERT(XML,'<Names><name>'
   + REPLACE(Name,' ', '</name><name>') + '</name></Name>>') AS xmlname
      FROM somenames
 SELECT
xmlname.value('/Names[1]/name[1]','varchar(100)') AS first_name,
xmlname.value('/Names[1]/name[2]','varchar(100)') AS last_name
 FROM Split Names
```

and also check the link below for reference

http://jahaines.blogspot.in/2009/06/converting-delimited-string-of-values.html

answered Feb 27 '13 at 9:23





Here is how I did this on a SQLite database:



SELECT SUBSTR(name, 1,INSTR(name, " ")-1) as Firstname, SUBSTR(name, INSTR(name, " ")+1, LENGTH(name)) as Lastname FROM YourTable;



Join Stack Overflow to learn, share knowledge, and build your career.









SELECT

SUBSTRING_INDEX(SUBSTRING_INDEX(rent, ' ', 1), ' ', -1) AS currency, SUBSTRING_INDEX(SUBSTRING_INDEX(rent, ' ', 3), ' ', -1) AS rent FROM tolets



edited Mar 12 '17 at 8:15 Fruchtzwerg

answered Mar 12 '17 at 5:54



That's MySQL, not MS SQL – AjV Jsy Apr 11 at 9:39

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up or sign in with



Facebook