

# Remove the last character in a string in T-SQL?

[Ask Question](#)

How do I remove the last character in a string in T-SQL ?

125

For example:



'TEST STRING'



to return:

28

'TEST STRIN'

sql



sql-server

tsql

edited Apr 17 '15 at 8:34



gotqn

21.3k

32

122

196

asked Aug 10 '09 at 20:01



Daveed

636

2

6

5

20 Answers

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

Email Sign Up

OR SIGN IN WITH



Google

Facebook



Home

PUBLIC

Stack Overflow

Tags

Users

.Jobs

[Learn More](#)

```

SET @String =
    CASE @String WHEN null THEN null
    ELSE (
        CASE LEN(@String) WHEN 0 THEN @String
        ELSE LEFT(@String, LEN(@String) - 1)
        END
    ) END

```

```

SELECT @String

```

edited Jul 9 '16 at 1:25



Shiroy

502 1 5 16

answered Aug 10 '09 at 20:02



AdaTheDev

109k 23 168 179

59 Alternatively: SELECT LEFT(YourColumnName, LEN(YourColumnName) - 1) FROM YourTable – [Kyle Ballard](#) Aug 10 '09 at 20:05

7 Note, this throws an error if your string is empty. – [gregmac](#) Jan 4 '13 at 21:33

3 Thanks for the null catch - ISNULL(LEFT(@String, LEN(@String) - 1), 'ErrMsg') will solve – [Volvox](#) Apr 18 '13 at 22:57

3 This also works set @String = STUFF(@String,1,1,'') – [BornToCode](#) Nov 12 '13 at 13:27

2 @Volvox it will still throw exception if the string is Empty String, i am modifying your answer to handle this scenario. – [Imran Rizvi](#) Oct 14 '15 at 14:03

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

Email Sign Up

OR SIGN IN WITH



Google

Facebook

95



causes you to have to repeat the same logic in the `len()` function. Duplicating the same logic becomes a mess. If this is the case then this is a solution worth noting. This example gets rid of the last unwanted comma. I finally found a use for the `REVERSE` function.

```
select reverse(stuff(reverse('a,b,c,d,'), 1, 1, ''))
```

answered Feb 2 '12 at 1:15



Bill Hoenig

951 6 2

6 Note that this code returns `NULL` if passed a string that is shorter than the delete range specified for `STUFF`. Wrap it in an `ISNULL` to get a different output value for the empty string case. – Rozwel May 2 '12 at 14:18

11 Nice, using this with an outer apply, with a for xml path("") to eliminate a trailing comma. awesome – Tracker1 Sep 27 '12 at 19:02

Also the same way as @Tracker1. Unlike anything involving `LEN()` this gracefully works (without repeating anything) for empty string (especially wrapped in `ISNULL()`) – gregmac Jan 4 '13 at 21:32

It is really remarkable how brutal SQL can be sometimes. This is incredible. – eouw0o83hf Apr 1 '13 at 21:10

Thanks, it helped me, in my case i had a last space so the first index is 2 `select reverse(stuff(reverse('a,b,c,d,'), 2, 1, ''))` – Arngue Jul 18 '18 at 7:55

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

Email Sign Up

OR SIGN IN WITH



Google

Facebook

answered Aug 10 '09 at 20:03



Adrien

2,525 17 17

@Adrien any idea why this seems to give the same result as  
 select substring('test string', 0, len('test  
 string')) ? – [Louis](#) Jan 23 '13 at 6:37

@Louis, the substring syntax is as follows: SUBSTRING (   
 expression ,start , length ) . Now, both queries return the  
 same because the numbering is 1 based, meaning that the  
 first character in the expression is 1. If start is less than 1, the  
 returned expression will begin at the first character that is  
 specified in expression. [Source](#) – [JS5](#) Feb 12 at 17:58

If your string is empty,

23

```
DECLARE @String VARCHAR(100)
SET @String = ''
SELECT LEFT(@String, LEN(@String) - 1)
```

then this code will cause error message 'Invalid length  
 parameter passed to the substring function.'

You can handle it this way:

```
SELECT LEFT(@String, NULLIF(LEN(@String)-1,-1))
```

It will always return result, and NULL in case of empty  
 string.

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

Email Sign Up

OR SIGN IN WITH



Google

Facebook

▲ If your column is `text` and not `varchar`, then you can use this:

7 `SELECT SUBSTRING(@String, 1, NULLIF(DATLENGTH(@String)-1,`



edited Oct 26 '12 at 13:34



Polyfun

7,780 4 27 37

answered May 22 '12 at 14:17



Behrens

1,025 5 27 50

▲ `select left('TEST STRING', len('TEST STRING')-1)`

7



answered Nov 20 '15 at 12:58



greg121

619 1 10 18

▲ If you want to do this in two steps, rather than the three of REVERSE-STUFF-REVERSE, you can have your list separator be one or two spaces. Then use RTRIM to trim the trailing spaces, and REPLACE to replace the double

5

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

Email Sign Up

OR SIGN IN WITH



Google

Facebook

However, this is not a good idea if your original string can contain internal spaces.

Not sure about performance. Each REVERSE creates a new copy of the string, but STUFF is a third faster than REPLACE.

also see [this](#)

edited May 23 '17 at 12:03



Community ♦

1 1

answered Oct 2 '12 at 13:19



Daryl

51 1 1



```
@result = substring(@result, 1, (LEN(@result)-1))
```

5



edited Aug 3 '17 at 12:22



Sagar V

9,024 7 28 59

answered Aug 3 '17 at 12:21



farrukh saleem

51 1 1

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

Email Sign Up

OR SIGN IN WITH



Google

Facebook

from  
t;

### [SQL Server Fiddle Demo](#)

edited Apr 17 at 12:15

answered Aug 15 '17 at 6:53



[shA.t](#)

13.3k 4 38 74

you can create function

2

```
CREATE FUNCTION [dbo].[TRUNCRIGHT] (@string NVARCHAR(max),
RETURNS NVARCHAR(max)
AS
BEGIN
    IF LEN(@string)<@len
        RETURN ''
    RETURN LEFT(@string, LEN(@string) - @len)
END
```

answered Apr 18 '15 at 22:39



[Mihail Katrikh](#)

71 1 9

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

Email Sign Up

OR SIGN IN WITH



Google

Facebook

answered Oct 19 '16 at 5:51

[Abhishek Jaiswal](#)

809 9 4

My answer is similar to the accepted answer, but it also check for Null and Empty String.

1

```
DECLARE @String VARCHAR(100)
```

```
SET @String = 'asdfsdf1'
```

```
-- If string is null return null, else if string is empty ,  
the end character
```

```
SET @String = Case @String when null then null else (case |  
@String else LEFT(@String, LEN(@String) - 1) end ) end
```

```
SELECT @String
```

answered Oct 14 '15 at 13:59

[Imran Rizvi](#)

5,098 8 42 88

Get the last character

1

```
Right(@string, len(@String) - (len(@String) - 1))
```

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

[Email Sign Up](#)

OR SIGN IN WITH

[Google](#)





Sam

11 1

I don't think this is what he was asking, though -- this might be useful to put in a comment, however. – [jimwise](#) Jan 24 '13 at 17:02

3 just Right(@string, 1). – [GoalBased](#) Apr 10 '14 at 21:14



The source text/var can be null or empty:

1

```
SELECT REVERSE(SUBSTRING(REVERSE(@a), 2, 9999))
```



answered May 2 at 13:36



David Roach

21 4



I love @bill-hoenig 's answer; however, I was using a subquery and I got caught up because the REVERSE function needed two sets of parentheses. Took me a while to figure that one out!

0



```
SELECT
```

```
-- Return comma delimited list of all payment reasons for
```

```
REVERSE(STUFF(REVERSE((
```

```
SELECT DISTINCT
```

```
CAST(CONVERT(varchar, r1.CodeID) + ' - ' +
```

```
FROM VisitReason r1
```

```
LEFT JOIN ReasonCode c ON c.ID = r1.ReasonID
```

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

Email Sign Up

OR SIGN IN WITH



Google



answered Sep 11 '12 at 16:18

[hurleystylee](#)

432 1 9 17

You may want to use an outer/cross apply for the composite portion of the query. I'm using this to get a set of flags for a parent item. – [Tracker1](#) Jan 4 '13 at 23:01



To update the record by trimming the last N characters of a particular column:

0



```
UPDATE tablename SET columnName = LEFT(columnName , LEN(co.
```

edited Oct 27 '15 at 6:34

[Azik Abdullah](#)

6,768 13 65 112

answered Oct 27 '15 at 6:16

[Manvendra\\_0611](#)

49 3



Try It :

0



```
DECLARE @String NVARCHAR(100)
SET @String = '12354851'
SELECT LEFT(@String, NULLIF(LEN(@String)-1,-1))
```

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

[Email Sign Up](#)

OR SIGN IN WITH



Google



▲  
0  
▼

```
declare @string varchar(20)= 'TEST STRING'  
Select left(@string, len(@string)-1) as Tada
```

output:

Tada  
-----  
TEST STRIN

answered Oct 27 '15 at 7:38



Migo  
123 10

▲  
0  
▼

Try this,

```
DECLARE @name NVARCHAR(MAX) SET @name='xxxxTHAMIZHMANI****'  
(len(@name)-8)) as UserNames
```

And the output will be like, THAMIZHMANI

answered May 7 '17 at 11:03



Thamizhmani  
9 3

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

Email Sign Up

OR SIGN IN WITH



Google

Facebook

-1

```
select
case when @x is null then @y
      when @y is null then @x
      else @x+', '+@y
end

go

declare @x varchar(20),@y varchar(20)
select @x='sam'
--,@y='john'
DECLARE @listStr VARCHAR(MAX)

SELECT @listStr = COALESCE(@x + ', ', '' ) +coalesce(@y+', ', '' )
SELECT left(@listStr,len(@listStr)-1)
```

edited May 15 '13 at 11:44



Taryn ♦

196k 47 299 359

answered May 15 '13 at 9:18



Julie

11

---

I think the existing answers already cover this method. – Sam  
Aug 15 '13 at 2:16

---

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

[Email Sign Up](#)

OR SIGN IN WITH



Google

