String replacement in batch file

Asked 10 years, 1 month ago   Active 1 year, 3 months ago   Viewed 197k times

▲

101

▼

🔖

20

🕐

We can replace strings in a batch file using the following command

```
set str="jump over the chair"
set str=%str:chair=table%
```

These lines work fine and change the string "jump over the chair" to "jump over the table". Now I want to replace the word "chair" in the string with some variable and I don't know how to do it.

```
set word=table
set str="jump over the chair"
??
```

Any ideas?

command-line   batch-file

	edited May 5 '10 at 10:48	asked May 5 '10 at 10:43
	Péter Török	Faisal
	106k 27 252 325	**3,554** 6 29 52

▲   possible duplicate of [how to replace string inside a bat file with command line parameter string](#)
🚩   which has much more informal answer (it has the same solutions as the answers here though) –
     Fr0sT Mar 18 '15 at 7:25 ✏️

---

1 ▲   now it's not important who was first, but it's good to link identical questions and point to the most
   🚩   irrefragable answer – Fr0sT Mar 25 '15 at 12:26

---

3 Answers

Active	Oldest	Votes

▲

78

▼

You can use !, but you must have the ENABLEDELAYEDEXPANSION switch set.

```
setlocal ENABLEDELAYEDEXPANSION
set word=table
set str="jump over the chair"
set str=%str:chair=!word!%
```

✔️

🕐

	answered May 5 '10 at 10:52
	Vicky
	11.8k 4 43 50

---

4 ▲   But what if `str` itself is from delayed expansion? `set str=!str:chair=!word!!` does not work
   🚩   correctly. – ImaginaryHuman072889 Oct 26 '18 at 12:41

▲

87

▼

↺

You can use the following little trick:

```
set word=table
set str="jump over the chair"
call set str=%%str:chair=%word%%%
echo %str%
```

The `call` there causes another layer of variable expansion, making it necessary to quote the original `%` signs but it all works out in the end.

answered May 5 '10 at 13:17

Joey
298k 75 615 632

▲

I like this solution, escaping strings is always problematic in batch files, ENABLEDELAYEDEXPANSION just adds another character to worry about. – Anders May 5 '10 at 19:43

---

9 ▲

Important thing regarding answer provided by **Joey**. That you need to put the code into batch file for it to work. If you just test it in command line, it will return unexpected `%"jump over the "word%%` . Just be aware, that code in batch files and in command line may produce different results. – dadhi Apr 23 '14 at 5:42

---

2 ▲

Building on dadhi's spot-on comment, the solution for command line is here: stackoverflow.com/questions/29944902/…. – Kenn Sebesta Jun 16 '17 at 18:46

---

2 ▲

Upvoting this answer because it works both ways, with the environment variable in either position, or in both the "before" and "after" positions: `set word=table` `set str="jump over the chair"` `call set str=%%str:chair=%word%%%` `echo %str%` `set word1=chair` `set word2=desk` `set str="jump over the chair"` `call set str=%%str:%word1%=%word2%%%` `echo %str%'` – Tom Warfield Sep 17 '17 at 17:51

▲

Please note "call" before the set to make it work – Mubashar Apr 1 '19 at 23:12

|

▲

-4

▼

↺

This works fine

```
@echo off
set word=table
set str=jump over the chair
set rpl=%str:chair=%word%
echo %rpl%
```

answered Feb 25 '19 at 13:31

Sérgio Maurício Trad
1

---

5 ▲

Sorry, it looks good, BUT it's wrong! It removes the word `chair` and append the word `table` , but it doesn't exchange the two words. Try to replace the word `over` with `under` and you get `jump the chairunder` – jeb Feb 25 '19 at 13:50