

## Variable Edit/Replace

Use the syntax below to edit and replace the characters assigned to a string variable.

### Syntax

```
%variable:StrToFind=NewStr%

%~[param_ext]$variable:Param
```

### Key

```
StrToFind      : The characters we are looking for
NewStr         : The chars to replace with (if any)
variable       : The environment variable
param_ext      : Any filename Parameter Extension
Param          : A command line parameter (e.g. 1)
```

This Edit/Replace syntax can be used anywhere that you would use the %variable% such as ECHOing the variable to screen or setting one variable = another

param\_ext cannot be %\* which typically represents a whole set of parameters, but this is easily worked around by setting a variable=%\*

"StrToFind" can **begin** with an asterisk, in which case it will replace all characters to the left of "StrToFind".  
By leaving NewStr blank, you can delete characters.

Using both an asterisk and NewStr=null effectively provides a left\$() or right\$() function.

### Examples:

The variable \_test containing 12345abcabc is used for all the following examples:

```
::Replace '12345' with 'Hello '
SET _test=12345abcabc
SET _result=%_test:12345=Hello %
ECHO %_result%           =Hello abcabc

::Replace the character string 'ab' with 'xy'
SET _test=12345abcabc
SET _result=%_test:ab=xy%
ECHO %_result%           =12345xycxyc

::Delete the character string 'ab'
SET _test=12345abcabc
SET _result=%_test:ab=%
ECHO %_result%           =12345cc

::Delete the character string 'ab' and everything before it
SET _test=12345abcabc
SET _result=%_test:*ab=%
ECHO %_result%           =cabc

::Replace the character string 'ab' and everything before it with 'XY'
SET _test=12345abcabc
SET _result=%_test:*ab=XY%
ECHO %_result%           =XYcabc

:: To remove characters from the right hand side of a string is
:: a two step process and requires the use of a CALL statement
:: e.g.

SET _test=The quick brown fox jumps over the lazy dog

:: To delete everything after the string 'brown'
:: first delete 'brown' and everything before it
SET _endbit=%_test:*brown=%
Echo We dont want: [%_endbit%]
```

```

::Now remove this from the original string
CALL SET _result=%_test:~_endbit%=%
echo %_result%

```

## Remove spaces from a text string

To delete space characters use the same syntax as above:

```
SET _no_spaces=%_some_var: =%
```

## Boolean Test "does string exist ?"

To test for the existence of a value we can use a temporary variable, delete the string we are looking for (if it exists) and then compare the two variables with [EQU](#)

Example: test for the existence of the string "London" in a variable containing text (that could be in any order) "Aberdeen, London, Edinburgh"

```

Set _cities="Aberdeen, London, Edinburgh"
:: Remove London if found
Set _dummy=%_cities:London=%
IF NOT %_dummy% == %_cities% (ECHO London was found.) ELSE (ECHO London was not found.)

```

## Finding items within the PATH environment variable

The %PATH% variable contains a list of folder names.

If you have a parameter containing a valid 'folder' this can be compared with the PATH variable.

This is done using the syntax: \$variable:parameter

Example

```

%PATH% = C:\Windows\system32;C:\WINNT;C:\utils\jdk\bin
batch parameter %1 = C:\utils\jdk\bin

```

To get the drive and Path

```
ECHO %~dp$PATH:1
```

This will either return "C:\utils\jdk\bin" or a NULL if the item is not found in the %PATH%

If the batch parameter was supplied as %2 then this would be: `ECHO %~dp$PATH:2`

This syntax can be applied where:

- The parameter is any valid parameter (%1 %2 %G) but it must contain a Full [Path](#) (not a pathname)
- The variable is %PATH% or any other variable that contains one or more Paths or pathnames separated by semicolons ;
- If nothing is found by the search, then this will return an empty string (NULL)

Be wary of using the syntax on this page to **modify** the PATH - the User path can be edited, but the System path remains read-only for most users.

*"A phony smile will never replace honest integrity" ~ Bob Martinelli*

## Related:

[SUBSTRING](#) of a variable :~

[PARAMETERS](#) - Filename Parameter Extensions

[PATHMAN](#) - This Resource Kit utility allows quick modification of both the system and user paths. Pathman can resolve many problems such as duplicate characters, and can improve performance by removing duplicate paths. For details see Pathman.wri in the resource kit.

**Petites Annonces Maroc**[www.avito.ma](http://www.avito.ma)

Déposez votre Annonce sur  
Avito.ma C'est facile, rapide et  
Gratuit !



AdChoices 



© Copyright [SS64.com](http://SS64.com) 1999-2012  
Some rights reserved