

Call command in Windows

The call command calls a batch program from another program without stopping the original batch program. The call command accepts the label as its target.

The call command calls a batch program from another program without stopping the original batch program. The call command accepts the label as its target.

Note : The call command is not valid at the command prompt, when it is used outside the script or executable file.

Call command in Windows

1. Syntax
2. Parameters
3. The batch parameter
4. Note
 1. Use batch parameter
 2. Use Label parameter
 3. Use the symbol (!) and navigation icon
 4. Perform a recursive call command
 5. Work with command extensions
5. For example

Syntax

```
call [Drive:] [Path] [] [: []]
```

Parameters

ParametersDescription [:] [] Specifies the location and name of the batch program you want to call. The *FileName* parameter is required and must have a .bat or .cmd extension. Specify any command line information required by the batch program. : Specify the *label* to which you want the batch program control to be transferred. Specifying the command line information will be transferred to the new object in the batch program, starting at : *Label*. /? Show help at the command prompt.

The batch parameter

The reference for batch scripts (% 0,% 1, .) is listed in the following table.

% * in batch script refers to all arguments (eg: % 1,% 2,% 3 .)

You can use the following optional syntax to replace batch parameters (% n):

Batch parameter	Description
% ~ 1	Expand % 1 and remove the surrounding quotation marks ("). % ~ f1 Expand % 1 to qualified path. % ~ d1 Only expand % 1 to drive letter. % ~ p1 Only expanded % 1 to path. % ~ n1 Only expand % 1 to a file name.
% ~ x1	Expand only % 1 file name extension. % ~ s1 Expand % 1 to a qualified path containing only short names. % ~ a1 Expand % 1 to file attribute. % ~ t1 Expand % 1 to the date and time of the file. % ~ z1 Expand % 1 to the size of the file. % ~ \$ PATH: 1 Search for the directories listed in the PATH environment variable and expand % 1 to the full name of the first directory found. If the environment variable name is not defined or the file is not found, this modifier will expand to an empty string.

The following table shows how you can combine modifiers with batch parameters to produce the following concatenation results:

Batch parameter with modifier	Description
% ~ dp1	Only expand % 1 to drive letter and path. % ~ nx1 Only expand % 1 to file name and extension. % ~ dp \$ PATH: 1 Search for the directories listed in the PATH environment variable for % 1, and then expand to the drive letter and path of the first directory found. % ~ ftza1 Expand % 1 to display the output similar to the dir command .

In the above examples, % 1 and PATH can be replaced with other valid values. The syntax % ~ is ended with a number of valid arguments. Cannot use modifiers % ~ with % *.

Note

Use batch parameter

The batch parameters can contain any information that you can convert to a batch program, including command line options, file names, batch parameters from % 0 to % 9 and variables (eg % baud%).

Use Label parameter

By using the call command with the Label parameter, you will create a new context for the batch file and pass control to the command after the specified label. The first time the executable file ends (that is, after jumping to label), a statement is returned after the call statement. The second time the executable file ends, the batch script will exit.

Use the symbol (!) and navigation icon

Do not use characters (!) and navigation symbols () with the call command.

Perform a recursive call command

You can create a batch program that calls itself. However, you must provide an exit condition. Otherwise, the original batch programs and subroutines can form non-stop loops.

Work with command extensions

If the command extension is enabled, the call command accepts the label as the target of the command. The syntax is exactly as follows:

```
call:
```

For example

To run **Checknew.bat** from another batch program, type the following command into the original batch program:

```
call checknew
```

If the original batch accepts two batch parameters and you want it to pass those parameters to **Checknew.bat**, type the following command in the original batch program:

```
call checknew% 1% 2
```

See more:

1. The command `nfsadmin` in Windows
2. The `ntbackup` command in Windows
3. `Nslookup` command in Windows

You finished reading the article "**Call command in Windows**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.