

# What is the difference between % and %% in a cmd file?



I recently included a line similar to this in a .cmd file:

67

```
for /f %%f in ('dir /b .\directory\*.sql') DO sqlcmd -b -o ".\directory\output\%%f.txt"
-i ".\directory\%%f"
```



21

Originally I had only used %f, and it would work fine when run on the command line, but not when run through the file. When I switched to %%f, it worked in the file. Just wondering what the difference is.

batch-file

cmd

asked Jan 24 '13 at 19:57



Tyler

555

1

5

11

## 2 Answers



(Explanation in more details can be found in [an archived Microsoft KB article](#).)

111

Three things to know:



1. The percent sign is used in batch files to represent command line parameters: %1 , %2 , ...
2. Two percent signs with any characters in between them are interpreted as a variable:



```
echo %myvar%
```

3. Two percent signs *without* anything in between (in a batch file) are treated like a single percent sign in a command (not a batch file): %%f

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

Email Sign Up

OR SIGN IN WITH



Google



```
FOR /f %f in ('dir /b .') DO somecommand %f
```

in a batch file, *rule 2* would try to interpret

```
%f in ('dir /b .') DO somecommand %
```

as a variable. In order to prevent that, you have to apply *rule 3* and escape the % with an second % :

```
FOR /f %%f in ('dir /b .') DO somecommand %%f
```

edited May 16 '18 at 8:22

answered Jan 24 '13 at 20:27



[marapet](#)

**42.6k** 9 128 151

**19** Is there an explanation for why I can get away with using just one percent sign in a FOR loop, when executed directly from a command prompt, and not from a batch file? I know about the DOS heritage, but we **can** use variables from a command line now. – [Olegs Jeremejevs](#) Oct 29 '13 at 11:58

**4** Also, make sure the variable is a single character. – [jiggunjer](#) Sep 2 '16 at 2:22

What if I want to issue command which has % in it e.g.: for /f %i in ('git log -1 --pretty=format:"%%H"') do set GIT\_COMMIT=%i – [Wakan Tanka](#) Jan 25 at 11:45



12



In DOS you couldn't use environment variables on the command line, only in batch files, where they used the % sign as a delimiter. If you wanted a literal % sign in a batch file, e.g. in an echo statement, you needed to double it.

This carried over to Windows NT which allowed environment variables on the command line, however for backwards compatibility you still need to double your % signs in a .cmd file.

answered Jan 24 '13 at 20:23



[Neil](#)

**44.9k** 8 45 63

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

Email Sign Up

OR SIGN IN WITH



Google

Facebook

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

Email Sign Up

OR SIGN IN WITH

 Google

Facebook 