



Quick Tip: Changing Default Directory of Git Bash

DECEMBER 9, 2014 / TUTORIAL / 19 COMMENTS

I don't know why this gave me so much trouble, but it did. When I installed **Git for Windows**, the default, Bash-like Windows terminal for **git**, it insisted that its home directory was /s (nowhere near my Documents folder in windows).

Here is a quick tip to make Git Bash change its home directory automatically (so you don't have to `cd /blah/blah/blah` every time you open it).

Open your Git for Windows client. Use vim to create a `.bashrc` file (this file gets automatically run every time the terminal is open):

```
vim .bashrc
```

Press 'i' to go into 'Insert' mode. Type in the following (change `<my/desired/home/dir>` to your new home directory).

Sign up for my newsletter

CATEGORIES

- Classroom



```
export HOME=/c/Users/shawn.hymel/Documents/GitHub  
  
s:\.bashrc [dos] (14:29 09/12/2014) 3,50 All  
-- INSERT --
```

Press 'esc' to exit "Insert" mode. Type ":wq" and press 'enter' to save and exit. Now, reload your Git terminal window, and you should be in a new home directory!

EDIT (12/9/14): Added export line to .bashrc file to update environment variable.

[BASH](#) [GIT](#) [QUICK TIPS](#) [TUTORIAL](#)

Share this Article



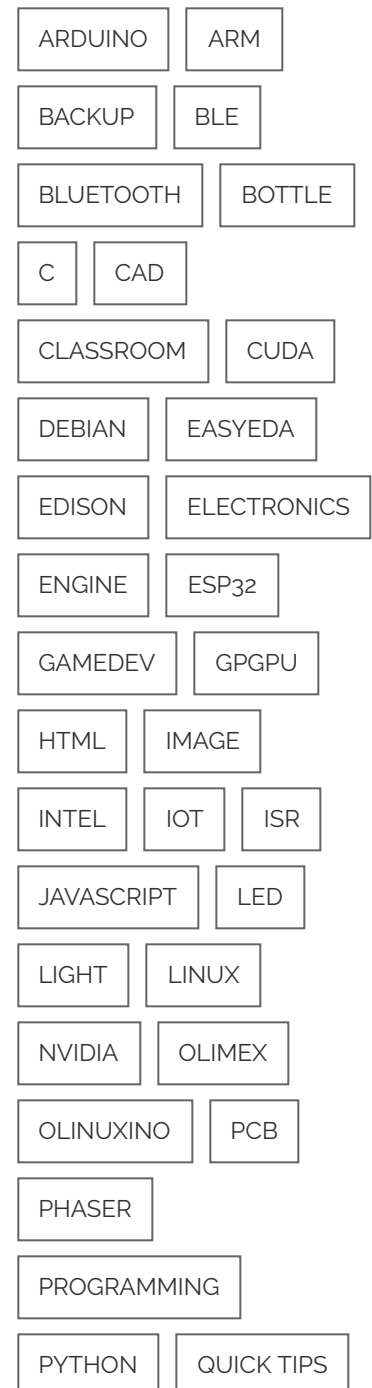
Previous Article

LED Circuit From Magnet
Video

Next Article

Creating a Custom Linux
Kernel for the Edison

TAGS

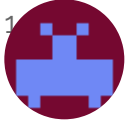


Author

Default Directory of Git Bash

WEBSOCKET

WIFI



Igor on June 8, 2016 at 5:42 pm [REPLY](#)

didn't work for me =/



ShawnHymel on June 10, 2016 at 7:22 pm

[REPLY](#)

Sorry to hear that. Did you make sure the home directory was typed out exactly as you intended (e.g. double check it in something like Windows Explorer)? I've done this trick a few times now, and it still works for me.



Jordan on June 22, 2016 at 12:03 am [REPLY](#)

didn't work for me either.

Now it shows line 1: syntax error near unexpected token `cd'



ShawnHymel on June 23, 2016 at 3:02 am

[REPLY](#)

The space between { and 'cd ...' is necessary. For some reason, the script won't work without it.



SHAWN
HYMEL



Vlas on July 22, 2016 at 9:24 am [REPLY](#)

Good, thanks! Really helpful. But anyway, it is not necessary to use vim 😊



ShawnHymel on July 22, 2016 at 2:54 pm

[REPLY](#)

True, but easy enough to do from within Git Bash 😊



The Bro on October 11, 2016 at 10:23 pm [REPLY](#)

Worked like a champ!



Jacques on March 24, 2017 at 5:24 pm [REPLY](#)

Thanks, it worked fine.



Roen on April 7, 2018 at 9:22 am [REPLY](#)

Thanks! It worked.



ShawnHymel on April 9, 2018 at 3:30 pm

[REPLY](#)



SHAWN
HYMEL



I'm late to the game, but why not:

```
export HOME=/c/  
cd
```



ShawnHymel on May 23, 2018 at 12:59 am

[REPLY](#)

I suppose that will work, too. More than one way to skin a cat 😊



B on June 2, 2018 at 3:34 am [REPLY](#)

Worked for me on Windows10. Thanks a ton!
Cheers!



M on June 12, 2018 at 5:24 pm [REPLY](#)

Worked on Win10, thanks!



M on June 6, 2019 at 3:43 pm [REPLY](#)

Did not work for me...



ShawnHymel on June 7, 2019 at 1:59 pm

[REPLY](#)



Charles on September 13, 2019 at 5:29 pm [REPLY](#)

Wonder is have a space in the file name will keep this from working? Where would you put the quotation marks?



ShawnHymel on September 16, 2019 at 3:44 pm [REPLY](#)

To work with spaces in the path, you need to enclose the whole path in single (') or double (") quotation marks:

`'/c/Program Files/Something'`

`"/c/Program Files/Something"`

You can also use the backslash (\) character to escape the space character:

`/c/Program\ Files/Something`

Leave a Comment

Your email address will not be published. Marked fields are required.



SHAWN
HYMEL



Name *

Email *

Website



I'm not a robot

reCAPTCHA
[Privacy](#) - [Terms](#)

SUBMIT



COPYRIGHT 2018 | SHAWN HYMEL