



SPONSORED BY



Change Cygwin Prompt

Asked 13 years, 1 month ago Modified 2 years, 2 months ago Viewed 29k times



18



8



The default cygwin prompt of `"user@computer path \n $"` is too long for me. I would like to keep the path.

I want it to become:

`path $`

Is there a config file I can modify to do this?

cygwin

Share Improve this question Follow

edited Jul 17, 2009 at 1:05

asked Jul 17, 2009 at 0:59



Cristian

343 ● 2 ● 4 ● 9

Possible duplicate of [Is it possible to change my terminal window prompt text?](#) – phucv Sep 8, 2018 at 9:08

[How to change the format of the bash prompt?](#) – phucv Sep 8, 2018 at 9:09

6 Answers

Sorted by:

Highest score (default) ▾



27



The `PS1` environment variable controls the prompt:

```
PS1='\w $ '
```

For more information on this and other prompt configuration topics, type `man bash` (assuming `bash` is your shell) and see the "PROMPTING" section.

To make this change permanent, edit your `~/.bashrc` file to add the above line.

Share Improve this answer Follow

edited Jul 17, 2009 at 1:07

answered Jul 17, 2009 at 1:01



Greg Hewgill



- 1 ~/.bashrc does not get executed for a login shell. update ~/.bash_profile instead. I use the following prompt string, which has some other useful information, not just the path: PS1='\[\e[32m\]t \[\e[33m\]w \[\e[31m\]!\[\e[0m\]\\$ ' – bobmcn Aug 26, 2009 at 20:16
- 3 Don't forget that normally .profile sources .bashrc, so that in effect, a login shell is initialized with the same stuff than a non-login shell *plus* what's in .profile. If that's the case, putting your new prompt in .bashrc kills two birds with one stone. – user290253 Jan 15, 2014 at 14:06



2



A login shell is one whose first character of argument zero is a -, or one started with the --login option. When bash is invoked as an interactive login shell, or as a non-interactive shell with the --login option, it first reads and executes commands from the file /etc/profile, if that file exists. After reading that file, it looks for ~/.bash_profile, ~/.bash_login, and ~/.profile, in that order, and reads and executes commands from the first one that exists and is readable. The --noprofile option may be used when the shell is started to inhibit this behavior. When an interactive shell that is not a login shell is started, bash reads and executes commands from ~/.bashrc, if that file exists.

So it depends...i don't use the --login, so i must add it to ~/.bashrc

Share Improve this answer Follow

answered Mar 18, 2013 at 9:53



jr00n
121 ● 1



2



Not sure why having less context is better than having more... The fact that there is a new line in the prompt means the length of the prompt should not be an issue, but try this:

```
PS1='\[\e[1;33m\]w\n\[\e[1;36m\]\$ \[\e[0m\] '
```

or

```
export PS1='\[\e[1;33m\]w\n\[\e[1;36m\]\$ \[\e[0m\] '
```

This gives you a coloured prompt:

```
/full/path/to/current/folder
$your command text here
```

That way, you always see your full folder context but still get a full line to input text. (I left out the customary space following the '\$' because it's coloured for clarity).

Colours are:

1. '/full/path/...' = yellow;
2. '\$' (on next line) = cyan;

3. 'your command text...' = light grey.

For those who DO want the 'user@hostname' context too:

PS1='[\e[1;32m\]\u[\e[1;35m\]@\e[1;32m\]\h \e[1;33m\]\w\n[\e[1;36m\]\\$\e[0m\]
,

or

```
export PS1='\[\e[1;32m\]\u\[\e[1;35m\]@\[\e[1;32m\]\h \[\e[1;33m\]\w\n\[\e[1;36m\]\$\[\e[0m\] '
```

This gives you a coloured prompt:

```
user@hostname /full/path/to/current/folder
$your command text here
```

This is my preference.

Colours are:

1. 'user' = (light) green;
2. '@' = pink;
3. 'hostname' = (light) green;
4. '/full/path/...' = yellow;
5. '\$' (on next line) = cyan;
6. 'your command text...' = light grey.

(No, there are no spelling mistakes in this post - Queen's English ;))

Share Improve this answer Follow

answered Jan 16, 2017 at 9:49

skeetastax
241 ● 2 ● 6

.bashrc didn't work for me. I added this to the end of /etc/profile and it worked:

1

```
export PS1="\[\e[33m\]\w\[\e[0m\] \$ "
```

I'm using Cygwin version 2.11.2 (latest version as of 2018-12-18).

Share Improve this answer Follow

answered Dec 18, 2018 at 21:23



Samuel
121 • 5

Put this in your ~/.bashrc. Gives a coloured prompt and keeps the status in a single line.

0

```
export PS1="\[\e]0;\w\a\]\n\[\e[32m\]\u@h \[\e[33m\]\w\[\e[0m\]\$ "
```

Share Improve this answer Follow

answered May 13, 2016 at 13:51



Sandeep

101 ● 1

I personally found that the other examples modified the prompt too much for my liking, so I found it more useful to determine where the current Cygwin prompt is set, what it looks like and then modify that as needed.

0

Turns out it's located in `/etc/bash.bashrc`, and looks like this:

```
PS1='\[\e]0;\w\a\]\n\[\e[32m\]\u@h \[\e[33m\]\w\[\e[0m\]\n\$ '
```

I simply wanted to add a timestamp to my prompt to make it more practical, so I modified the above to add the current time to the very beginning in dark grey, and then added the string to my personal `~/.bashrc` file (located in `/home/<Username>/.bashrc`):

```
PS1='\[\e]0;\w\a\]\n\[\e[90m\]\t \[\e[32m\]\u@h \[\e[33m\]\w\[\e[0m\] \n\$ '
```

You can use a generator like [this](#) to help more visually learn the syntax of the PS1 prompt. [Here](#) is a comprehensive list of all the colours you can make use of in the process.

Share Improve this answer Follow

answered Jun 23, 2020 at 0:41



Hashim Aziz

10.6k ● 30 ● 83 ● 143