


How do I type on the next line in the Terminal?

 askubuntu.com/questions/226204/how-do-i-type-on-the-next-line-in-the-terminal



Web-E

18.1k1111 gold badges4545 silver badges8989 bronze badges

12

As Web-E explains the most direct way to do what you want with two different commands, I thought I'd show that there are a number of ways to execute multiple commands or to continue commands onto another line without immediately executing them.

Continuing long commands:

1) The most common way to construct one long command is to enter your commands, then use a backslash `\`, press return, and then Bash will provide another prompt for you instead of executing the command. This secondary prompt is called PS2 and waits for your input:

```
find /home/mike/Downloads -type f -iname '*.jpg' \  
>
```

You can keep on adding backslashes and pressing return as long as you want, as long as you think the overall command will make sense.

You can cancel this secondary prompt with the usual `Ctrl + C`.

2) Bash recognises some commands such as for loops (for i in....) and the prompt will appear immediately; just as it will if you miss a quotation mark off a command:

```
apt-cache search 'libgimp*  
>
```

Multiple Commands:

3) As Lxns1ck notes, you can separate commands with semicolons:

```
which vlc; whereis vlc
```

```
/usr/bin/vlc  
vlc: /usr/bin/vlc /etc/vlc /usr/lib/vlc /usr/bin/X11/vlc /usr/share/vlc /usr/share/man/man1/vlc.1.gz
```

4) Or you can use the ampersand `&&` to join two commands:

```
./configure && make
```

answered Dec 6 '12 at 17:09

user76204

4

You can press the ENTER key after each line and if the command is not terminated (multiline commands like `for` loops for example), the terminal will wait for you to enter the rest of the command. If the command is terminated, it will be executed and you enter next command after, no problem.

If you are copying the commands from a tutorial, you can copy the whole group of commands and paste it directly in the terminal and it will work.

edited Dec 6 '12 at 16:57

answered Dec 6 '12 at 16:51



laurent

5,39911 gold badge2020 silver badges2323 bronze badges

1

When it's a set of commands you expect to be using more than once, you should put them in a bash *script file*. For instance,

```
xrandr --newmode "1280x960_80.00" 139.25 1280 1368 1504 1728 960 963 967 1008 -hsync +vsync
xrandr --verbose --addmode VGA1 "1280x960_80.00"
xrandr --output HDMI1 --off --output LVDS1 --mode 1366x768 --pos 1280x512 --rotate normal --output DP1
--off --output VGA1 --mode "1280x960_80.00" --pos 0x0 --rotate normal
sleep 3
xfce4-panel -r
```

is nothing you'd like to type ever again, but I happen to need this particular sequence of commands very often. So it goes in a file called `1280x980-2head.sh`, you can create that with any editor of your choice. To make that script *executable*, you put the line `#!/bin/bash` in front of everything (a shebang), and set the execution-permission: `chmod +x 1280x980-2head.sh`. Then, just typing `./128<tab>` (auto-completion) and `<enter>` executes all the commands.

answered Dec 6 '12 at 21:07



leftaroundabout

59933 silver badges1313 bronze badges

0

You can put everything in brackets; for example:

```
(sudo add-apt-repository ppa:foo/bar  
sudo apt-get update  
sudo apt-get install my-package)
```

The commands will be executed one after the other in a sub-shell.

If you don't want them to be executed in a sub-shell, you can put a semicolon between a command and the other; for example:

```
sudo add-apt-repository ppa:foo/bar; sudo apt-get update; sudo apt-get install my-package
```

Instead, if you put "&&" between each commands, the command that follows the "&&" will be executed **only** if the previous command has finished with no error, so you don't have to use it to concatenate commands; use semicolons instead. Conversely, if you put "||", the following command will be executed only if the first one exits with an error. Example: (the "which" command checks if a program exists, and if doesn't exits with an error)

```
which apt-get && echo "No error" || echo "Error"    # Will print "No error"  
which jhdhsd && echo "No error" || echo "Error"    # Will print "Error"
```

answered Dec 13 '12 at 21:51



[Depau](#)

4611 silver badge99 bronze badges

Your Answer

Sign up or log in



Sign up using Email and Password

Post as a guest

Required, but never shown

Not the answer you're looking for? Browse other questions tagged [command-line](#) or [ask your own question](#).
