

skillcrush

MAC TERMINAL

cheatsheet

GET TO KNOW YOUR TERMINAL

A command line interface (or the command line) is a text-only way of browsing and interacting with your computer. On Macs, the command line is called the Terminal.

To access your Mac Terminal:

1. Open up your Applications folder.
2. Scroll down to the 'Utilities' folder, then open that up.
3. Double-click Terminal.
4. You're in!

```
user - bash - 80x24

Last login: Wed Jun 12 18:59:35 on ttys000
*~$ cd sites
*~/sites$ ls
BarWarriors      david            old
DailyWorth3      dw-newsletters  python
atw80            gardenaccents   ruby
bar_warriors     gardenaccents.zip  ruby-class
bar_warriors2     hello.php        rubygems-2.0.3
bootcamp         ladies-hack-day  skillcrush
bunnybot         mml             skillcrush-api
capelin          mml2            todomvc
capelin 1        mnn             untitled folder
charles          mymathlab       youtube
corbis           newsletter
*~/sites$
```

COMMON COMMANDS

`cd / cd .. / cd - / cd ~`

The `cd` command allows you to navigate in and out of directories or folders.

`cd` gets you into a folder:

```
$~ cd
```

`cd ..` gets you out of a folder into the parent directory:

```
$~ cd ..
```

`cd -` returns you to the directory you were just in:

```
$~ cd -
```

`cd ~` returns you to your home directory:

```
$~ cd ~
```

`pwd`

`pwd` stands for "print working directory." Using the `pwd` command allows you to see what folder you are in via your Terminal.

```
$~ pwd
```

`open .`

The `open .` command lets you open up the directory in Finder.

```
$~ open .
```

`ls`

The `ls` command lists all of the folders and files in your current directory:

```
$~ ls
```

mkdir

The *mkdir* command allows you to make new directories or folders. Don't forget to give your folder a name, like so:

```
$~ mkdir sites
```

FUN COMMANDS

date

The *date* command returns today's date:

```
$~ date
```

cal

The *cal* command, followed by the year, returns a calendar for that year:

```
$~ cal 2016
```

whois

The *whois* command is the Yellow Pages of domains. It allows you to view the registration details of a given domain, such as the name, address, and phone number of the registrant. Larger domain holders tend to block their registration details from view:

```
$~ whois google.com
```

ping

The `ping` command allows you to ping a web server from your computer. Usually, you would want to do this to see if a website is up and running. In order to check, just write *ping url.com*:

```
$~ ping google.com
```

If the site is up, you will see something like this:

```
PING google.com (74.125.226.225): 56 data bytes
64 bytes from 74.125.226.225: icmp_seq=0 ttl=55 time=3.341 ms
64 bytes from 74.125.226.225: icmp_seq=1 ttl=55 time=6.808 ms
```

You can make it stop pinging by typing:

```
$~ ^C
```

(Or you can press control-c)

say

The `say` command prompts your Mac to speak out loud in its native voice whatever you write after the `say` prompt:

```
$~ say Hello My Name is Alf
```

MORE COMMANDS

The commands shared above are just the beginning of what you can do with Terminal! Here are some more that you can play with on your own time.

sudo

The *sudo* command stands for “superuser do.” By preceding your command with this command, you grant yourself admin rights over everything on your computer:

```
$~ sudo ls /
```

touch

The *touch* command allows you to create new files. Remember to include your file’s name and proper file extension:

```
$~ touch myrubyapp.rb
```

open [-e | -a | -t]

The open commands allows you to open files or applications from terminal.

Use *open -e filename* to open a file with TextEdit (the native Mac text editing application)

```
$~ open -e mytext.txt
```

Use *open -a applicationname* to open an application like TextMate or Adobe Photoshop.

```
$~ open -a TextMate
```

When an application name has spaces in it, identify those spaces with a \ like so:

```
$~ open -a Adobe\ Photoshop\ CS
```

Use *open -t* to open with the default text editor (if different than TextEdit).

```
$~ open -t myfile.rb
```

rm -Rf DANGER

The `rm -Rf` command allows you to delete specific files. In order to delete a file, you must include its file path:

```
$~ rm -Rf /Users/username/Sites/bunnybot/myfile.rb
```

If you aren't sure of the file's file path, you can drag the file into Terminal and the Terminal will automatically sense its filepath. But be careful! Once you've hit enter, that file is gone baby gone.

man

The `man` command allows you to read the manual page for most commands:

```
$~ man cd
```

The man page includes lots of fun things like a description of the command, a list of flags you can use and the history of the command. To move to the next page of the manual, press the spacebar. To move to the previous page, press the B key. To exit out of the manual, press the Q key.

⌘k (command k)

The *command-k* command clears your terminal. Use this to declutter!

defaults write com.apple.Finder AppleShowAllFiles YES

This command allows you to reveal all of the Apple hidden files. Use it to see files like your `.bash_profile` and other fun ones like that. Make sure to include the *killall Finder* command in order to restart Finder so that your changes will appear.

```
$~ defaults write com.apple.Finder AppleShowAllFiles YES
```

```
$~ killall Finder
```

Here's a place where capitalization matters! Depending on what operating system you're using, you may need to use lowercase letters for Finder in the first line:

```
$~ defaults write com.apple.finder AppleShowAllFiles YES
```

```
$~ killall Finder
```

```
defaults write com.apple.Finder AppleShowAllFiles NO
```

This command allows you to hide all of the Apple hidden files. Make sure to include the *killall Finder* command in order to restart Finder so that your changes will appear.

```
$~ defaults write com.apple.Finder AppleShowAllFiles NO
```

```
$~ killall Finder
```

Again, depending on your operating system, capitalization may matter. If the above command doesn't work for you, try this one:

```
$~ defaults write com.apple.finder AppleShowAllFiles NO
```

```
$~ killall Finder
```