

General Advice

💡 **Use the tab-completion:** Tab and double Tab

💡 **Use the history:** ↑ and ↓ and `ctrl+r`

CMD = any command

TXT = any text; escape special chars with \ or " or ''

FILE = any file name including the path (full/relative) to this file

PATH = a *FILE* or a directory accordingly

[] = an argument to a *CMD* which is optional

Get Help

whatis *CMD* display one-line descriptions of *CMD*

CMD -h (or -?, -help, --help, --usage) displays the help message of *CMD*

man *CMD* displays the manual pages of *CMD* ⓘ

info *CMD* displays the info documents of *CMD* ⓘ

help *CMD* display information about the builtin *CMD*

apropos *TEXT* search the man pages for *TEXT*

groups *USER* shows all groups of *USER* (default is own)

Show File Content

head *FILE* print the first 10 lines of each *FILE*

head -n *X* *FILE* print the first *X* lines

tail *FILE* print the last 10 lines of each *FILE*

tail -n *X* *FILE* print the last *X*

less *FILE* display *FILE* on a page-by-page basis ⓘ

| **less** display STDIN on a page-by-page basis ⓘ

cat *FILE* prints whole *FILE*

cat *FILE1* *FILE2*> *FILE3* cats *FILE1* and *FILE2* and prints them to *FILE3*

Rediretcs & Pipes & Subshell

CMD> *FILE* prints standard output of *CMD* to *FILE*, replacing it

CMD>> *FILE* prints standard output of *CMD* to *FILE*, appending it

CMD1 | **CMD2** 'pipes' standard out of *CMD1* as standard input to *CMD2*

CMD> out.log runs *CMD* in background, prints standard out to out.log & **2**> err.log & out.log and standard error to err.log

Searching

grep *TXT* *FILES* search for pattern *TXT* in *FILES* (line-wise)

grep -r *TXT* *PATH* searches all files in *PATH* recursively

grep -v *TXT* *PATH* show lines not containing *TXT*

grep -c *TXT* *PATH* count matching lines instead of showing them

find [*PATH*] -name "*TXT*" finds all files named *TXT* in *PATH* (default .)

find [*PATH*] -iname "***TXT***" finds all files matching *TXT* case insensitive

Operate On Filesystem

pwd prints current directory

ls [*PATH*] list directory contents

ls -lart [*PATH*] long + hidden + reverse order + sort last modified

cd changes to HOME directory

cd *PATH* changes to *PATH* directory

mkdir *PATH* creates a directory

mkdir -p *PATH* creates also parents as needed

cp *FILE* *PATH* copies *FILE* to *PATH*

cp -r *PATH1* *PATH2* cp recursively whole directory *PATH1* to *PATH2*

mv *PATH1* *PATH2* moves file or dir *PATH1* to *PATH2*

rm *FILE* removes *FILE*

rm -r *PATH* removes files and dirs under *PATH* recursively

ln -s *PATH1* creates link of *PATH* with same name

ln -s *PATH1* *TXT* creates link of *PATH* with new name *TXT*



By sjuenemann

Published 18th September, 2015.

Last updated 11th May, 2016.

Page 1 of 2.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>

Helpers

alias <i>TXT</i> =' <i>CMD</i>	abbreviate <i>CMD</i> with <i>TXT</i>
★	alias ..='cd ..'
★	alias l='ls -lah'
<i>CMD</i> wc -l	prints number of lines <i>CMD</i> generates as output
<i>CMD</i> sort	sorts output of <i>CMD</i> alphabetical
<i>CMD</i> sort -n -k 3	sorts output of <i>CMD</i> numerical based on 3rd column
<i>CMD</i> uniq	only prints unique lines of <i>CMD</i> output

Archives & Compression

gunzip <i>FILE.gz</i>	uncompresses the <i>FILE.gz</i> in-place
unzip <i>FILE.zip</i>	uncompress and extracts the ZIP archive <i>FILE.zip</i>
tar -xvzf <i>FILE.tgz</i>	extracts + uncompress (gzip) the tarball <i>FILE.tgz</i>
tar -xvf <i>FILE.tar</i>	extracts the tarball <i>FILE.tar</i>
tar -cvzf <i>FILE.tgz</i> <i>PATH</i>	archives and compresses <i>PATH</i> into <i>FILE.tgz</i>

Permissions

chmod	change the permissions of FILE (or directory) to OCTAL
OCTAL	seperately for user, group and world by summing:
<i>FILE</i>	<ul style="list-style-type: none"> ⊕4 - read (r) ⊕2 - write (w) ⊕1 - execute (x)
	Example: chmod 750 == rwx for owner, rx for group, none for world

Process Control

top	displays realtime information about the top cpu processes
ps	report snapshot of processes (in current shell)
ps -lfu <i>USER</i>	show all processes of <i>USER</i> in details
kill <i>PID</i>	terminate the process with the process ID <i>PID</i>

Shortcuts

Ctrl + c	Interrupt/Kill whatever you are running (SIGINT)
Ctrl + z	Suspends current task (SIGTSTP).
fg + Enter	: resume process
bg + Enter	: continue process in background
Shift + PageUp	Scroll terminal window up
Shift + PageDown	Scroll terminal window down

Shortcuts (cursor movement)

Ctrl + a / Home	Go to beginning of the line
Ctrl + e / End	Go to the End of the line
Ctrl + f / →	Forward one character
Ctrl + b / ←	Backward one character
Ctrl + a	Delete current character
← Backspace	Delete previous character
Shift + Insert	Paste text

Wildcards

*	replaces arbitrary number of characters
?	replaces exactly one character
[abc]	replaces exactly for a, b or c
[a-z]	replaces for any char from a to z

Navigation (less, man, ...)

h	display help message
q	quit
space-bar PageDown	one page down
b PageUp	one page up
q	quit
/ <i>TXT</i>	search forward for <i>TXT</i>
? <i>TXT</i>	search backward for <i>TXT</i>
n	repeat previous search
N	repeat previous search in reverse direction
G	Go to last line
g	Go to first line



By sjuenemann

Published 18th September, 2015.

Last updated 11th May, 2016.

Page 2 of 2.

Sponsored by [Readable.com](https://readable.com)

Measure your website readability!

<https://readable.com>