

# The Linux Command Line Bootcamp

CHEATSHEET FOR COLT STEELE’S UDEMY COURSE  
(CREATED BY QIUSHI YAN)

## - Getting Help

Display the manual page for a command  
`man [command] ...`

The **man** pages are a built-in format of documentation. Each man page contains the synopsis of a command syntax. For instance, a simplified synopsis for the **sort** looks like  
`sort [-n] [-h] [-k=number] [file] ...`

Brackets indicate an **optional** argument, possibly followed by parameter types. An ellipsis ... means more than one of the preceding operand is allowed. This means **sort** accepts optional argument -n, -h and -k, and -k expects a number, and we can provide more than one file to sort with.

Shortcuts for navigating man pages.

Q	quit man page
B/F	go back/forward a page
/PATTERN	search for a pattern
H	viewing all shortcuts

For certain shell builtins without a manual page entry, the **help** command can be used to generate a help page.

## - Navigation

The root directory / and home directory ~ are two important directories in a user’s system.

print working directory

`pwd`

list contents of a directory, default to current

`ls [dir]`

Common options for `ls` include

- **-a**: print files that begin with .
- **-l** include permissions, file size and modification time
- **-h** print human redable sizes

move into another directory

`cd [dir]`

Use `cd ..` to move up one level.

Refer to the root directory and user home directory with / and ~ respectively.

## - Manipulating Files and Directories

create files

`touch [file] ...`

create new directories

`mkdir [dir] ...`

The **-p** option automatically makes parent directories as needed.

copy files and directories

Copy a single file or directory item1 to item2

`cp [item1] [item2]`

Copy multiple items into one directory

`cp [file]... [dir]`

move and rename files

Move or rename the file or directory item1 to item2

`mv [item1] [item2]`

Move items from one directory to another

`mv [item].. [dir]`

delete files and directories

`rm [file] ...`

By default, rm can only delete files and **empty** directories.

.....

Options for **rm**

Option	Long	Description
<b>-i</b>	--interactive	prompt before removal
<b>-r</b>	--recursive	allow removing non-empty directories
<b>-f</b>	--force	do not prompt

.....

**Recipe:** remove all files with confirmation

`rm -i [dir]/*`

edit files with nano

nano is a text-based editor. `nano file` will edit the provided file name with nano, or create a new file if it does not exist. `nano +line file` will open the file at a specific line.

nano shortcuts

ctrl+O	write out
ctrl+S	save
ctrl+X	exit nano
ctrl+W	search forward
ctrl+\	replace
M+\\, M+/ <td>move to the first / last line</td>	move to the first / last line
ctrl+A, ctrl+E	move to the start / end of a line

.....

Edit /etc/nanorc for furthur configuration.

## - File Manipulation Cont.

concatenate and print the contents of files

`cat file1 file2` outputs file1 first and immediately after prints the contents of file2. For long files, **less** displays the contents one page at a time.

`tac` prints files in reverse order (last line first). **rev** reverse lines characterwise.

.....

Options for **cat**

Option	Long	Description
<b>-n</b>	--number	number output lines
<b>-s</b>	--squeeze-black	suppress repeated black lines
<b>-A</b>	--show-all	show non-printable characters such as tabs and line endings

print first / last parts of files

The **head** command prints the first ten lines of a file, and the **tail** command prints the last ten lines. The number of lines can be adjusted with the **-n** option, or simply **-[number]**.

`head -n5 file`

`tail -5 file`

The **-f** option of **tail** views file contents in real time. This is useful for monitoring log files.

print line, word, byte counts

`wc [file] ...` prints newline, word, byte counts for each file and a total line of all files

To limit the output, use

- **-w**: print word counts
- **-l**: print line counts
- **-m**: print character counts
- **-C**: print byte counts

.....

**Recipe:** count total lines of **.js** files

`wc -l *.js`

sort lines of fines

By default, `sort file` print each line from the specified file, sorted ini alphabetical order. It can also merge multiple files into one sorted whole via `sort file1 file2 ...`.

.....

Options for **sort**

Option	Long	Description
<b>-n</b>	--numeric-sort	compare based on string numerical value
<b>-h</b>	--human-numeric-sort	compare based on human readable numbers (e.g., 2k 1G)
<b>-k</b>	--key=KEYDEF	sort via a key
<b>-r</b>	--reverse	sort in reverse order
<b>-u</b>	--unique	sort unique values only

.....

**Recipe:** find the top 10 biggest files inside a directory

`ls -lh [dir] | sort -rhk5 | head -10`

Alternatively, we can use the **du** command for showing file sizes.