

# Shell Cheat Sheet

Patrick Bucher

2017-05-18

## Contents

<b>Commands</b>	<b>2</b>
Files and Directories . . . . .	2
File Attributes . . . . .	3
Searching Files . . . . .	3
File Transfer . . . . .	3
Archives . . . . .	3
Text . . . . .	3
Spelling . . . . .	4
Document Preparation . . . . .	4
Text Editors . . . . .	4
Miscellaneous . . . . .	4
Date and Time . . . . .	5
Math . . . . .	5
Documentation . . . . .	5
Printing . . . . .	5
Programming . . . . .	6
Shell Scripting . . . . .	6
Scripting Languages . . . . .	6
Building, Debugging . . . . .	6
System Administration . . . . .	6
Processes . . . . .	7
User Administration . . . . .	7
Networking . . . . .	7
Printers . . . . .	7
Hashing and Crypto . . . . .	7
<b>Shell Programming</b>	<b>8</b>
Output Redirection . . . . .	8
<b>Variables</b>	<b>8</b>

# Commands

## Files and Directories

- **cd**: change the working directory
  - **cd**: change to home directory
  - **cd . .**: change to parent directory
  - **cd /home/joe**: change to directory using a absolute path
  - **cd docs/letters**: change to directory using a relative path
- **cp**: copy files and directories
  - **cp [source] [destination]**: copy source file to destination
  - **-f**: force overwrite an existing file
  - **-i**: prompt before overwrite
  - **-n**: don't overwrite an existing file
  - **-r**: copy a directory recursively
- **dd**: convert and copy a file
  - **dd if=[in] of=[out] bs=[n]**: copy from in to out, n bytes at a time
- **du**: estimate file space usage
  - **-h**: print sizes in human readable format
  - **-s**: display only a total for each argument
- **ln**: make links between files
  - **ln [target] [link name]**: create a (hard) link to target
  - **-s**: make a symbolic link instead of a hard link
- **ls**: list directory contents
  - **-a**: also display hidden entries (starting with ".")
  - **-h**: print human readable sizes (in combination with **-l**)
  - **-l**: long listing format (**ll** is short for **ls -l**)
  - **-r**: reverse order
  - **-R**: list files recursively
  - **-t**: sort by modification time, newest first
- **mkdir**: make directories
  - **make foo bar**: create the directories "foo" and "bar"
- **mv**: move (rename) files
  - **mv [source] [destination]**
  - **-f**: force overwrite an existing file
  - **-i**: prompt before overwrite
  - **-n**: don't overwrite an existing file
- **pwd**: print name of current/working directory
- **rm**: remove files or directories
  - **-d**: remove empty directories
  - **-f**: force overwrite an existing file
  - **-i**: prompt before every removal
  - **-r**: delete a directory recursively

- `touch`: change file timestamps – or create a new file
  - `-c`: don't create a new file

## **File Attributes**

- `basename`: strip directory and suffix from filenames
- `cleannname`: clean a path name
- `chmod`: change file mode bits
- `chown`: change file owner and group
- `file`: determine file type
- `fsize`: print file information
- `mtime`: print file information

## **Searching Files**

- `find`: search for files in a directory hierarchy
- `locate`: find files by name
- `updatedb`: update a database for `locate`

## **File Transfer**

- `curl`: transfer a URL
- `ftp`: File Transfer Protocol client
- `rsync`: a fast, versatile, remote (and local) file-copying tool
- `scp`: secure copy (remote file copy program)
- `sftp`: secure file transfer program
- `wget`: the non-interactive network downloader

## **Archives**

- `ar`: create, modify, and extract from archives
- `tar`: an archiving utility
- `zip`: package and compress (archive) files
- `unrar`: unpack RAR archives
- `unzip`: list, test and extract compressed files in a ZIP archive

## **Text**

- `cat`: concatenate files and print on the standard output
- `cut`: remove sections from each line of files

- `fmt`: simple optimal text formatter
- `fold`: wrap each input line to fit in specified width
- `diff`: compare files line by line
- `freq`: print histogram of character frequencies
- `grep`: print lines matching a pattern
- `head`: output the first part of files
- `join`: join lines of two files on a common field
- `look`: display lines beginning with a given string
- `read`: copies one line from a file to standard output
- `sed`: stream editor for filtering and transforming text
- `sort`: sort lines of text files
- `split`: split a file into pieces
- `strings`: print the strings of printable characters in files
- `tail`: output the last part of files
- `tee`: read from standard input and write to standard output and files
- `tr`: translate or delete characters
- `wc`: print newline, word, and byte counts for each file
- `uniq`: report or omit repeated lines

## Spelling

- `aspell`: interactive spell checker
- `hunspell`: spell checker, stemmer and morphological analyzer

## Document Preparation

- `troff`: the troff processor of the groff text formatting system

## Text Editors

- `ed`: line-oriented text editor
- `emacs`: GNU project Emacs editor
- `vi`: screen oriented (visual) display editor based on `ex`
- `vim`: Vi IMproved, a programmers text editor
  - `vim -p [files]`: open multiple files in tabs

## Miscellaneous

- `acpi`: show battery status and other ACPI information
- `ascii`: interpret ASCII characters
- `chroot`: run command or interactive shell with special root directory

- `cmp`: compare two files byte by byte
- `dmesg`: print or control the kernel ring buffer
- `eval`: construct command by concatenating arguments
- `expr`: evaluate expressions
- `hexdump`: display file contents in hexadecimal, decimal, octal, or ascii
- `halt`: halt, power-off or reboot the machine
- `history`: manipulate the history list
- `lspci`: list all PCI devices
- `pwgen`: generate pronounceable passwords
- `shutdown`: halt, power-off or reboot the machine (see `halt`)
- `unicode`: interpret unicode characters
- `xrandr`: primitive command line interface to RandR extension
- `yes`: output a string repeatedly until killed

## Date and Time

- `cal`: display a calendar
- `date`: print or set the system date and time

## Math

- `bc`: an arbitrary precision calculator language (interactive)
- `dc`: reverse-polish desk calculator (interactive)
- `factor`: factor numbers
- `hoc`: interactive floating point language
- `primes`: list prime numbers
- `rand`: generate pseudo-random bytes

## Documentation

- `apropos`: search the manual page names and descriptions
- `man`: an interface to the on-line reference manuals
- `type`: write a description of command type
- `whatis`: display one-line manual page descriptions

## Printing

- `cancel`: cancel jobs
- `lpr`: print files

## **Programming**

### **Shell Scripting**

- `alias`: define or display aliases
- `break`: exit from `for`, `while`, or `until` loop
- `echo`: display a line of text
- `export`: set the export attribute for variables
- `getopts`: command-line parsing for shell scripts
- `printf`: format and print data
- `sleep`: delay for a specified amount of time
- `test`: check file types and compare values
- `xargs`: build and execute command lines from standard input

### **Scripting Languages**

- `awk`: pattern-directed scanning and processing language
- `perl`: the Perl language interpreter
- `python`: an interpreted, interactive, object-oriented programming language
- `sh`: shell, the standard command language interpreter

### **Building, Debugging**

- `gcc`: GNU project C and C++ compiler
- `gdb`: the GNU Debugger
- `make`: GNU make utility to maintain groups of programs
- `mk`: maintain (make) related files
- `strip`: discard symbols from object files
- `valgrind`: a suite of tools for debugging and profiling programs
- `yacc`: parser generator

### **System Administration**

- `df`: report file system disk space usage
- `fdisk`: manipulate disk partition table
- `fsck`: check and repair a Linux filesystem
- `mkfs`: build a Linux filesystem
- `mount`: mount a filesystem
- `su`: run a command with substitute user and group ID
- `sudo`: execute a command as another user
- `systemctl`: control the `systemd` system and service manager

- `uname`: print system information

## Processes

- `bg`: run jobs in the background
- `kill`: terminate a process
- `killall`: terminate a process by name
- `nice`: run a program with modified scheduling priority
- `nohup`: run a command immune to hangups, with output to a non-tty
- `ps`: report a snapshot of the current processes
- `top`: display Linux processes (interactive)

## User Administration

- `groupadd`: create a new group
- `groupdel`: delete a group
- `groupmems`: administer members of a user's primary group
- `groupmod`: modify a group definition on the system
- `groups`: display current group names
- `id`: print real and effective user and group IDs
- `passwd`: change user password
- `useradd`: create a new user or update default new user information
- `userdel`: delete a user account and related files
- `usermod`: modify a user account
- `users`: print the user names of users currently logged in

## Networking

- `ip`: show/manipulate routing, devices, policy routing and tunnels
- `ping`: send ICMP ECHO\_REQUEST to network hosts

## Printers

- `lpadmin`: configure cups printers and classes

## Hashing and Crypto

- `base32`: base32 encode/decode data and print to standard output
- `base64`: base64 encode/decode data and print to standard output
- `gpg`: OpenPGP encryption and signing tool

- `md5sum`: compute and check MD5 message digest
- `sha1sum`: compute and check SHA1 message digest
- `sha256sum`: compute and check SHA256 message digest
- `sha512sum`: compute and check SHA512 message digest
- `sum`: checksum and count the blocks in a file

## Shell Programming

### Output Redirection

```
wc -l * | sort -n -r  
wc -l >std-out 2>std.err
```

### Variables

- `$IFS`: internal field separator