

# LINUX AND ROS CHEAT SHEET

AUTHOR: YIZE WANG  
LAST UPDATED: MAY 5, 2020

## System Info

<b>env</b>	print environment variables
<b>date</b>	print system date and time
<b>cal</b>	print current month calendar
<b>uptime</b>	print system uptime
<b>w</b>	print online users
<b>whoami</b>	print current user
<b>finger \$USER</b>	print information about <b>\$USER</b>
<b>uname -a</b>	print kernel information
<b>cat /proc/cpuinfo</b>	print cpu information
<b>cat /proc/meminfo</b>	print memory information
<b>man \$COMMAND</b>	print user manual of <b>\$COMMAND</b>
<b>df</b>	print disk usage
<b>du</b>	print directory space usage
<b>free</b>	print memory and swap usage
<b>whereis \$APP</b>	print locations of <b>\$APP</b>
<b>which \$APP</b>	print print executable file of <b>\$APP</b>

## Compression

<b>tar cf \$FILE.tar \$FILES</b>	convert <b>\$FILES</b> into <b>\$FILE.tar</b>
<b>tar xf \$FILE.tar</b>	extract files from <b>\$FILE.tar</b>
<b>tar czf \$FILE.tar.gz \$FILES</b>	compress <b>\$FILES</b> into <b>\$FILE.tar.gz</b> using Gzip
<b>tar xzf \$FILE.tar.gz</b>	extract files from <b>\$FILE.tar.gz</b> using Gzip
<b>gzip \$FILE</b>	compress <b>\$FILE</b> and rename it as <b>\$FILE.gz</b>
<b>gzip -d \$FILE.gz</b>	decompress <b>\$FILE.gz</b> back to <b>\$FILE</b>

## Network

<b>ip address</b>	print all internet protocol addresses
<b>ping \$HOST</b>	ping <b>\$HOST</b> and print results
<b>whois \$DOMAIN</b>	print information about <b>\$DOMAIN</b>
<b>dig \$DOMAIN</b>	print DNS of <b>\$DOMAIN</b>
<b>dig -x \$HOST</b>	reverse lookup <b>\$HOST</b>
<b>wget \$FILE</b>	download <b>\$FILE</b>

## Terminator

<b>Ctrl+Alt+T</b>	launch a new terminal
<b>Ctrl+C</b>	kill the current process
<b>Ctrl+Z</b>	suspend the current process
<b>fg</b>	resume the suspended process in foreground
<b>bg</b>	resume the suspended process in background
<b>Ctrl+D</b>	log out of the current session
<b>Ctrl+W</b>	erase one word in the current line
<b>Ctrl+U</b>	erase the whole current line
<b>Ctrl+R</b>	reverse search in the previous commands
<b>!!</b>	execute the last command
<b>exit</b>	log out of the current session
<b>Ctrl+Shift+E</b>	split the window vertically vertically
<b>Ctrl+Shift+O</b>	split the window horizontally

## Package

<b>apt-get update</b>	synchronize package index files from sources
<b>apt-get upgrade</b>	install latest versions of installed packages
<b>apt-get install \$PACKAGE</b>	install <b>\$PACKAGE</b>
<b>dpkg -i \$PACKAGE.deb</b>	install a Debian package <b>\$PACKAGE.deb</b>
<b>./configure</b>	configure building settings
<b>make</b>	build the program from source code
<b>make install</b>	install the program

## Secure Shell (SSH)

<b>ssh \$USER @ \$HOST</b>	connect to <b>\$HOST</b> as <b>\$USER</b>
<b>ssh \$IP_ADDRESS</b>	connect to <b>\$IP_ADDRESS</b>
<b>ssh -p \$PORT \$USER @ \$HOST</b>	connect to <b>\$HOST</b> on <b>\$PORT</b> as <b>\$USER</b>
<b>ssh-copy-id \$USER @ \$HOST</b>	add the key to <b>\$HOST</b> as <b>\$USER</b>

## Searching

<b>grep \$PATTERN \$FILE</b>	search for <b>\$PATTERN</b> in <b>\$FILE</b>
<b>grep -r \$PATTERN \$DIR</b>	recursively search for <b>\$PATTERN</b> in <b>\$DIR</b>
<b>\$COMMAND   grep \$PATTERN</b>	search for <b>\$PATTERN</b> in <b>\$COMMAND</b> 's output
<b>locate \$FILE_NAME</b>	find all files whose name contain <b>\$FILE_NAME</b>

## Git

<b>git clone \$URL</b>	clone the repository from <b>\$URL</b>
<b>git status</b>	print current branch status <b>\$BRANCH</b>
<b>git branch \$BRANCH</b>	create a new branch named <b>\$BRANCH</b>
<b>git checkout \$BRANCH</b>	switch to the branch named <b>\$BRANCH</b>
<b>git merge \$BRANCH</b>	combine <b>\$BRANCH</b> into the current one
<b>git fetch</b>	download all history from GitHub
<b>git merge</b>	combine remote branches into local branch
<b>git push</b>	upload all local branch commits to GitHub
<b>git pull</b>	update local branch from GitHub
<b>git log</b>	list version history for current branch
<b>git log --follow \$FILE</b>	list version history for <b>\$FILE</b>
<b>git show \$COMMIT</b>	output content changes of <b>\$COMMIT</b>
<b>git add \$FILE</b>	stage <b>\$FILE</b>
<b>git commit -m "\$MESSAGE"</b>	commit staged file with <b>\$MESSAGE</b>
<b>git reset \$FILE</b>	reset <b>\$FILE</b>
<b>git reset --hard</b>	reset all uncommitted changes
<b>git clean -fd</b>	recursively force remove unstaged files

**Variable Declaration** VARIABLE stands for a variable whose name is VARIABLE. For example, FILE means a file named FILE.  
**sudo** sudo means super user do, elevated right granted

## File Commands

<b>ls</b>	list contents of files and directories
<b>ls -a</b>	list hidden files and directories
<b>cd \$DIR</b>	change working directory to <b>\$DIR</b>
<b>cd</b>	change working directory to home
<b>mkdir \$DIR</b>	create a directory named <b>\$DIR</b>
<b>pwd</b>	print working directory
<b>rm \$FILE</b>	remove <b>\$FILE</b>
<b>rm -r \$DIR</b>	remove <b>\$DIR</b>
<b>rm -f \$FILE</b>	force remove <b>\$FILE</b>
<b>rm -rf \$DIR</b>	force remove <b>\$DIR</b>
<b>cp \$FILE1 \$FILE2</b>	copy <b>\$FILE1</b> to <b>\$FILE2</b>
<b>cp -r \$DIR1 \$DIR2</b>	copy <b>\$DIR1</b> to <b>\$DIR2</b>
<b>mv \$FILE1 \$FILE2</b>	move <b>\$FILE1</b> to <b>\$FILE2</b>
<b>ln -s \$FILE \$LINK</b>	create symbolic link <b>\$LINK</b> to <b>\$FILE</b>
<b>touch \$FILE</b>	create <b>\$FILE</b>
<b>cat \$FILE</b>	view content of <b>\$FILE</b>
<b>cat &gt; \$FILE</b>	write input into <b>\$FILE</b>
<b>more \$FILE</b>	print contents of <b>\$FILE</b>
<b>head \$FILE</b>	print the first 10 lines of <b>\$FILE</b>
<b>tail \$FILE</b>	print the last 10 lines of <b>\$FILE</b>

## This image shows a blank sheet of white paper with horizontal blue ruling lines. The lines are evenly spaced and run across the width of the page. There are approximately 20 lines visible. On the left side, there is a vertical margin line, creating a narrow left margin. The paper appears to be from a notebook or a standard ruled document.

<b>roscore</b>	invoke the core of ros
<b>roslaunch</b> <u>\$PACKAGE</u> <u>\$EXECUTABLE</u>	run <u>\$EXECUTABLE</u> in <u>\$PACKAGE</u>
<b>roslaunch</b> <u>\$PACKAGE</u> <u>\$LAUNCHFILE</u>	launch <u>\$LAUNCHFILE</u> in <u>\$PACKAGE</u>

[illegible]

## This image shows a blank sheet of white paper with horizontal blue ruling lines. The lines are evenly spaced and run across the width of the page. There are approximately 20 lines visible. The paper has a slight shadow on the right side, suggesting it's part of a bound notebook or folder.

ubuntu mono font for commands
tab auto-completion
tar explanation
is git reset unstage or reset?
simtime true