Bash Cheat Sheet

General

> **man [command]**	Display the `man` pages for the `[command]`
> **[command] –help**	Display information about how to use the `[command]`
> **[command] -h**	Same as `--help`
> **exit**	Logout of the current session

System

> **date**	Display the current date and time
> **cal**	Display the month's calendar
> **uptime**	Display the current system uptime
> **whereis [command]**	Display possible locations of `[command]`
> **which [command]**	Display which application will be run by `[command]`
> **lsb_release -a**	Display the Ubuntu version
> **uname -r**	Display the kernel version
> **uname -a**	Display all kernel information
> **df**	Display disk usage
> **du**	Display directory space usage
> **shutdown**	Shutdown the Ubuntu system

Hardware Information

> **dmesg**	Display bootup messages
> **cat /proc/cpuinfo**	Display information about the CPU
> **cat /proc/meminfo**	Display information about the memory
> **free**	Display memory and swap usage
> **lshw**	Display hardware configuration info
> **lsblk**	Display block devices info
> **lspci -tv**	Display tree-diagram of PCI devices
> **lsusb -tv**	Display tree-diagram of USB devices
> **dmidecode**	Display BIOS hardware info
> **hdparm -i /dev/[disk]**	Display disk data information of `[disk]`
> **hdparm -tT /dev/[disk]**	Display disk read speed test of `[disk]`
> **badblocks -s /dev/[disk]**	Display unreadable blocks test of `[disk]`

Package management

> **apt update**	Refresh package list
> **apt upgrade**	Upgrade all packages
> **apt dist-upgrade**	Upgrade Ubuntu version
> **apt autoremove**	Remove all obsolete packages
> **apt install [package]**	Install `[package]`
> **apt remove [package]**	Remove `[package]`
> **apt search [package]**	Search for `[package]` in package list
> **dpkg -i [package].deb**	Manually install `[package]` using a .deb file

Directory navigation

> **pwd**	Show current directory
> **ls**	List the contents of the current directory (use the option `-al` to get all contents and more detailed information)
> **cd [dir]**	Change the current working directory to `[dir]`
> **cd /**	Change the current working directory to the root directory
> **cd ~**	Change the current working directory to your home directory
> **cd ./[subdir]**	Change the directory to the sub directory `subdir` of your current one
> **cd ..**	Change the current working directory to the parent directory
> **cd -**	Change the current working directory to the previous directory

File management

> **mkdir [dir]**	Create the directory `[dir]`
> **touch [file]**	Create `[file]`
> **cp [source] [target]**	Copy `[source]` to `[target]`
> **cp -r [source] [target]**	Copy `[source]` to `[target]` and create directory for `[target]`
> **cat [file]**	Outputs the contents of `[file]`
> **cat >[file]**	Writes the standard input into `[file]`
> **head [file]**	Outputs the first 10 lines of `[file]`
> **tail [file]**	Outputs the last 10 lines of `[file]`
> **tail -f [file]**	Continues to output lines as `[file]` grows
> **rm [file]**	Deletes `[file]`
> **rm -r [dir]**	Deletes `[dir]` and contained elements
> **rsync -a [directory] [backup]**	Sync the contents of `[directory]` with the directory `[backup]`

Users and groups

> **w**	Display logged in users and activity
> **who**	Display who you are logged in as
> **id**	Show active user details
> **last**	Show last system logins
> **finger [user]**	Display information about `[user]`
> **usermod**	Modify user information
> **adduser [user]**	Add `[user]` as a new user
> **groupadd [group]**	Add `[group]` as a new user group
> **usermod -aG [group] [user]**	Add user `[user]` to group `[group]`
> **userdel [user]**	Delete the user `[user]`

Bash variables

> **[variable]="[value]"**	Assign **[value]** to *[variable]* (creates the varaible, if it doesn't exist)
> **echo $[variable]**	Display the value of *[variable]*
> **set**	Display list of all bash variables and functions

Priviliges

> **sudo [command]**	Run `[command]` as superuser
> **sudo -k**	Forget superuser password
> **su**	Switch shell user to superuser
> **passwd**	Change the password of the current user

File permission

> **chmod [permissions] [file]**	Changes the permissions of `[file]` to `[permissions]` Permissions can be set for the **u**ser (owner of the file), the user **g**roup the file belongs to, **o**ther groups, or **a**ll. Possible permission are **r**ead, **w**rite, or e**x**ecute. Use the following syntax to set permissions: **(ugoa)=(w-)(r-)(x-)**. (**-** means the permission is not given.)
> **chown [owner] [file]**	Changes the owner of `[file]` to `[owner]`
> **chgrp [group] [file]**	Changes the user group of `[file]` to `[group]`

Searching

> **grep [pattern] [file]**	Search for pattern `[pattern]` in `[file]`
> **grep -r [pattern] [dir]**	Search recursively for pattern `[pattern]` in `[dir]`
> **[command] | grep [pattern]**	Search for pattern `[pattern]` in the output of `[command]` (`|` is a so called pipe operator and is used to forward the output of one command to the next command)
> **locate [file]**	Find all instances of `[file]`
> **find [dir] -name [file]**	Find all instances of `[file]` in `[dir]` or sub directories
> **pgrep [pattern]**	Search all named processes for `[pattern]`

System services

> **start [service]**	Start the job for `[service]`
> **restart [service]**	Restart the job for `[service]`
> **status [service]**	Check the status of the job for `[service]`
> **stop [service]**	Stop the job for `[service]`

Process management

> **ps**	Display active process snapshot
> **pstree**	Display processes as a tree
> **pmap**	Display process memory usage
> **htop**	Display all running processes (use `top` if not installed)
> **kill [pid]**	kills the process with the process id `[pid]`
> **killall [process]**	Kills all process with the name `[process]`
> **pkill [pattern]**	Kills all process matching `[pattern]`
> **lsof**	Display list of files opened by processes
> **nohup [command]**	Execute `[command]` as a process in background

Network

> **hostname**	Display system hostname
> **ifconfig**	Show network information
> **iwconfig**	Show wireless network information
> **sudo iwlist scan**	Scan for wireless networks
> **ping [host]**	Send ping to `[host]` and display the results
> **whois [domain]**	Display whois information for `[domain]`
> **dig [domain]**	Display DNS information for `[domain]`
> **dig -x [host]**	Reverse lookup `[host]`
> **wget [file]**	Download `[file]`
> **wget -c [file]**	Continue stopped download of `[file]`
> **scp [file] [host]**	Copy `[file]` to the directory `[host]` on a different machine
> **rsync -a [/location] [/backup/]**	Sync the contents of a location with the backup directory

Firewall

> **ufw enable**	Turn on the firewall
> **ufw disable**	Turn off the firewall
> **ufw default allow**	Allow all connections by default
> **ufw default deny**	Deny all connections by default
> **ufw status**	Display current status and rules
> **ufw allow [port]**	Allow traffic on `[port]`
> **ufw deny [port]**	Block `[port]`
> **ufw deny from [ip]**	Block traffic from `[ip]`

SSH login

> **ssh [user]@[host]**	Connect to `[host]` as `[user]` using ssh
> **ssh [host]**	Connect to *[host]* via port 22
> **ssh -p [port] [user]@[host]**	Connect to `[host]` as `[user]` using a non-default *[port]*

Shortcuts

ctrl + c	Halt the current command
ctrl + d	Logout of the current session
ctrl + w	Erase the word before the cursor
ctrl + u	Cut part of the line before the cursor
ctrl + k	Cut part of the line after the cursor
ctrl + r	Bring up a recent command
up arrow / down arrow	Cycle throw the last executed commands of the session
ctrl + shift + c	Copy marked section of text (marked using the mouse)
ctrl + shift + v	Paste copied text
tab	Autocomplete an input (doesn't work for everything, but is useful for paths for example)