



July 2020 by Tobias Günther

Command Line Cheat Sheet

Our cheat sheet explains the essential tasks on the command line.
 Download it for free.

For many, the command line belongs to long gone days: when computers were controlled by typing mystical commands into a black window; when the mouse possessed no power. But for many use cases, the command line is still absolutely indispensable!

Our cheat sheet not only features the most important commands. On the back, it also explains some tips & tricks that make working with the CLI a lot easier.



Download the Cheat Sheet

Get 8 of our most popular Cheat Sheets in one handy ZIP!

[Download Now for Free](#)





COMMAND LINE CHEAT SHEET

presented by Tower - the best Git client for Mac and Windows

DIRECTORIES	FILES	SEARCH
<pre>\$ pwd</pre> Display path of current working directory	<pre>\$ rm <file></pre> Delete <file>	<pre>\$ find <dir> -name "<file>"</pre> Find all files named <file> inside <dir> (use wildcards [*?] to search for parts of filenames, e.g. "file.*")
<pre>\$ cd <directory></pre> Change directory to <directory>	<pre>\$ rm -r <directory></pre> Delete <directory>	<pre>\$ grep "<text>" <file></pre> Output all occurrences of <text> inside <file> (add -i for case-insensitivity)
<pre>\$ cd ..</pre> Navigate to parent directory	<pre>\$ rm -f <file></pre> Force-delete <file> (add -r to force-delete a directory)	<pre>\$ grep -rI "<text>" <dir></pre> Search for all files containing <text> inside <dir>
<pre>\$ ls</pre> List directory contents	<pre>\$ mv <file-old> <file-new></pre> Rename <file-old> to <file-new>	
<pre>\$ ls -la</pre> List detailed directory contents, including hidden files	<pre>\$ mv <file> <directory></pre> Move <file> to <directory> (possibly overwriting an existing file)	<h2>NETWORK</h2> <pre>\$ ping <host></pre> Ping <host> and display status
<pre>\$ mkdir <directory></pre> Create new directory named <directory>	<pre>\$ cp <file> <directory></pre> Copy <file> to <directory> (possibly overwriting an existing file)	<pre>\$ whois <domain></pre> Output whois information for <domain>
<h2>OUTPUT</h2> <pre>\$ cat <file></pre> Output the contents of <file>	<pre>\$ cp -r <directory1> <directory2></pre> Copy <directory1> and its contents to <directory2> (possibly overwriting files in an existing directory)	<pre>\$ curl -O <url/to/file></pre> Download <file> (via HTTP[S] or FTP)
<pre>\$ less <file></pre> Output the contents of <file> using the less command (which supports pagination etc.)	<pre>\$ touch <file></pre> Update file access & modification time (and create <file> if it doesn't exist)	<pre>\$ ssh <username>@<host></pre> Establish an SSH connection to <host> with user <username>
<pre>\$ head <file></pre> Output the first 10 lines of <file>	<h2>PERMISSIONS</h2> <pre>\$ chmod 755 <file></pre> Change permissions of <file> to 755	<pre>\$ scp <file> <user>@<host>:/remote/path</pre> Copy <file> to a remote <host>
<pre>\$ <cmd> > <file></pre> Direct the output of <cmd> into <file>	<pre>\$ chmod -R 600 <directory></pre> Change permissions of <directory> (and its contents) to 600	<h2>PROCESSES</h2> <pre>\$ ps ax</pre> Output currently running processes
<pre>\$ <cmd> >> <file></pre> Append the output of <cmd> to <file>	<pre>\$ chown <user>:<group> <file></pre> Change ownership of <file> to <user> and <group> (add -R to include a directory's contents)	<pre>\$ top</pre> Display live information about currently running processes
<pre>\$ <cmd1> <cmd2></pre> Direct the output of <cmd1> to <cmd2>		<pre>\$ kill <pid></pre> Quit process with ID <pid>
<pre>\$ clear</pre> Clear the command line window		

30-day free trial available at [www.git-tower.com](#)



The best Git Client for Mac & Windows





COMMAND LINE CHEAT SHEET

presented by Tower - the best Git client for Mac and Windows

GETTING HELP	THE "CTRL" KEY	HOME FOLDER
On the command line, help is always at hand: you can either type <code>man <command></code> or <code><command> --help</code> to receive detailed documentation about the command in question.	Various keyboard shortcuts can assist you when entering text: Hitting CTRL+A moves the caret to the beginning and CTRL+E to the end of the line. In a similar fashion, CTRL+K deletes all characters after and CTRL+U all characters in front of the caret. Pressing CTRL+L clears the screen (similarly to the clear command). If you should ever want to abort a running command, CTRL+C will cancel it.	File and directory paths can get long and awkward. If you're addressing a path inside of your home folder though, you can make things easier by using the <code>~</code> character. So instead of writing <code>cd ~/Users/your-username/projects/</code> , a simple <code>cd ~/projects/</code> will do. And in case you should forget your user name, <code>whoami</code> will remind you.
<h2>FILE PERMISSIONS</h2> On Unix systems, file permissions are set using three digits: the first one representing the permissions for the owning user, the second one for its group, and the third one for anyone else. Add up the desired access rights for each digit as following: 4 – access/read (r) 2 – modify/write (w) 1 – execute (x) For example, 755 means "rwx" for owner and "rx" for both group and anyone. 740 represents "rwx" for owner, "r" for group and no rights for other users.	<h2>THE "TAB" KEY</h2> Whenever entering paths and file names, the TAB key comes in very handy. It autocompletes what you've written, reducing typos quite efficiently. E.g. when you want to switch to a different directory, you can either type every component of the path by hand: <pre>\$ cd ~/projects/acmedesign/docs/</pre> ...or use the TAB key (try this yourself): <pre>\$ cd ~/pr[TAB]jects/ac[TAB]medesign/d[TAB]ocs/</pre> In case your typed characters are ambiguous (because "ac" could point to the "acmedesign" or the "actionscrip" folder), the command line won't be able to autocomplete. In that case, you can hit TAB twice to view all possible matches and then type a few more characters.	<h2>OUTPUT WITH "LESS"</h2> The less command can display <i>and paginate</i> output. This means that it only displays one page full of content and then waits for your explicit instructions. You'll know you have less in front of you if the last line of your screen either shows the file's name or just a colon (:). Apart from the arrow keys, hitting SPACE will scroll one page forward, b will scroll one page backward, and q will quit the less program.
<h2>COMBINING COMMANDS</h2> If you plan to run a series of commands after another, it might be useful to combine them instead of waiting for each command to finish before typing the next one. To do so, simply separate the commands with a semicolon (;) on the same line. Additionally, it is possible to execute a command only if its predecessor produces a certain result. Code placed after the && operator will only be run if the previous command completes successfully, while the opposite operator only continues if the previous command fails. The following command will create the folder "videos" only if the cd command fails (and the folder therefore doesn't exist): <pre>\$ cd ~/videos mkdir ~/Videos</pre>	<h2>THE ARROW KEYS</h2> The command line keeps a history of the most recent commands you executed. By pressing the ARROW UP key, you can step through the last called commands (starting with the most recent). ARROW DOWN will move forward in history towards the most recent call. Bonus tip: Calling the history command prints a list of all recent commands.	<h2>DIRECTING OUTPUT</h2> The output of a command does not necessarily have to be printed to the command line. Instead, you can decide to direct it to somewhere else. Using the > operator, for example, output can be directed to a file. The following command will save the running processes to a text file in your home folder: <pre>\$ ps ax > ~/processes.txt</pre> It is also possible to pass output to another command using the (pipe) operator, which makes it very easy to create complex operations. E.g., this chain of commands will list the current directory's contents, search the list for PDF files and display the results with the less command: <pre>\$ ls grep ".pdf" less</pre>

30-day free trial available at [www.git-tower.com](#)



The best Git Client for Mac & Windows



Download the Cheat Sheet

Get 8 of our most popular Cheat Sheets in one handy ZIP!

[Download Now for Free](#)

We make Tower

...the best Git client for Mac and Windows.



We help over 100,000 users in companies like Apple, Google, Amazon, Twitter, and Facebook to easily & productively work with the Git version control system.

[Try it 30 days for free!](#)

Read next

...popular & related articles.

- > [Coming Up on the Roadmap \(2022\)](#)
- > [Git Cheat Sheet](#)
- > [Diff Tools on macOS](#)