

>_ Basic Bash



Last Update 2021-06-26 | [Twitter](#) | [Im_tavo](#) |

Basic Shortcuts

- (TAB) **complete**: Attempt to perform completion on the text before point. [\[more\]](#)
- (C-j) **accept-line**: Send a newline (`\n`)
- (C-c) **interrupt (kill)**: The current foreground process running in the terminal. [\[more\]](#)
- (C-d) **end-of-file (EOF)**: If there are no chars on the line, Readline returns EOF
- (C-z) **suspend**: stop the program execution. `fg` to resume, `j` `obs -l` to list and `kill <id>` to close it [\[more\]](#)
- (M-.) **insert-last-argument**: Move 'backwards' through the previous arguments
- (C-p) **previous-history**: Move 'backward' through the history list, fetching the previous command.
- (C-n) **next-history**: Move 'forward' through the history list, fetching the next command.
- (C-r) **reverse-search-history**: Search 'backward' through the history. This is an incremental-search.
- (C-s) **forward-search-history**: Search 'forward' through the history. Important! You need to disable the xon first to use it (`stty -ixon`)
- (C-a) **beginning-of-line**: Move to the start of the line.
- (C-e) **ending-of-line**: Move to the end of the line
- (M-b) **backward-word**: Move backward a word
- (C-b) **backward-char**: Move backward a character.
- (M-f) **forward-word**: Move forward a word
- (C-f) **forward-char**: Move forward a character.
- (C-u) **unix-line-discard**: Kill 'backward' from the cursor to the beginning of the current line.
- (C-k) **kill-line**: Kill 'forward' from the cursor to the end of the current line.
- (M-BS) **delete-backward-word**: Kill the word behind point (BS = backspace)
- (M-d) **delete-forward-word**: Kill from point to the end of the current word

Basic Shortcuts (cont)

- (C-l) **clear-screen**: Clear the screen, reprinting the current line at the top.
- (C-/) **undo**: Incremental undo, separately remembered for each line.
- (C-y) **yank**: Yank the top of the kill-ring into the buffer at point. [\[more\]](#)
- (M-y) **yank-pop**: Rotate the kill-ring, and yank the new top. You can only do this if the prior command is `yank` or `yank -pop`.
- (C-s) **transmit-off (XOFF)**: Stops the data flow from the keyboard [\[more\]](#)
- (C-q) **transmit-on (XON)**: Enable the data flow and release all the info contained [\[more\]](#)

(C-): Control key = `Ctrl`

(M-): Meta key [`⌘`] = `Alt`, [`⌘`] = `Opt` (check Notes)

It is a loose convention that (C-) operate on `chars` while (M-) operate on `words`. [\[more\]](#)

'Cut' and 'paste' are more recent jargon for 'kill' and 'yank'.

Notes

For Firefox users: Cheathography uses a preload option to load the default. To enable it. Write this in the URL bar `about: config`, search option and set it `true`

You can find more info in this link [\[more\]](#)

For Mac users: The Meta key is set to the `ESC`, but you can set it the terminal and go to

`Terminal > Preferences > Basic Default > Keyboard >`
If you need more information, read this tutorial [\[more\]](#)

Spanish Keyboard Mac users: There are some users that say that the above can disable some keys like `~#~}~\|`

For Windows user There are some ways to use the bash shortcuts in the native powershell, check this [\[Article\]](#) to see how to use Bash shortcuts

I really loved doing this cheatsheet, I expect that it could help you

More Info

gnu.org: Bash Reference Manual

ss64.com: syntax-keyboard

howtogeek.com: keyboard-shortcuts...

catern.com: Terminals are weird

catonmat.net: Bash emacs editing mode

github.com: awesome-cheatsheets

cheatography.com: Bash shortcuts, full-list