

## SHELL QUICK REFERENCE CARD

Frequently used SHELL commands – Version v1.3 December 2015

In this Reference Card, special characters are used as a placeholder for specific, non-alphabetical keys.  $\langle M \rangle$  denotes the meta-key, usually bound to  $\langle \text{alt} \rangle$  or  $\langle \text{esc} \rangle$  while  $\wedge$  is referred to the Controllkey (CTRL).

### Movements

$\wedge a$   $\wedge e$  ..... jump to begin, end of the line  
 $\wedge u$   $\wedge k$  ..... cut to beginning, end of the line  
 $\wedge f$   $\wedge b$  ..... move one *char* forward, backward  
 $\langle m \rangle f$   $\langle m \rangle b$  .... jump one word forward, backward  
 $\langle m \rangle \leftarrow$  or  $\wedge w$  ... cut text backward until space  
 $\langle m \rangle d$  ..... cut one word forward until space  
 $\wedge y$  ..... paste most recently cut out text

### History

Prompt all executed history events: **history**, clear entire history event cache: **history -c**

### Shortcuts

$\wedge p$  ..... fetch the previous command from the history list  
 $\langle m \rangle .$  ..... print last argument(s) to line  
 $\wedge r$   $\wedge s$  ..... search history backward, forward (incremental search)  
 $\langle m \rangle p$   $\langle m \rangle n$  .... search backward, forward using non-incremental search  
 $\langle m \rangle <$   $\langle m \rangle >$  ... move to the first, last line in history

### History-Commands

Example command is: **cd foo/bar**

**!!** ..... runs previous executed command (also known as ‘bang bang’)  
 $\Rightarrow$  **sudo !!: sudo cd foo/bar**  
**!\$** ..... repeats last *arg* of prev command  
**!n !-n** ..... executes command *n*, last *n*th command

**!*str* !?*str*?** ... runs last executed command starting with, containing *str*  
 $\Rightarrow$  **!cd: cd foo/bar**  
**!!:opt** ..... filter last command by designators or modifiers  
 $\Rightarrow$  **echo !cd:q: echo 'cd foo/bar'**  
**!#** ..... entire command line typed so far  
**!!:gs/old/new/**  
substitutes new for old  
 $\Rightarrow$  **quick substitution, replacing  $str_1$  with  $str_2$ :  $\wedge str_1 \wedge str_2$  ( $\Rightarrow$   $\wedge$  literally)**

### Options

you may append as many options as you want, e. g. you could do a: **!! : \$ : p** (prints least argument of last executed history command)

### Event Designators

**0** the first word in a line (usually command or programs name)  
**n** the *n*th word in a line  
 **$\wedge$**  the first *arg* (the second word) in a line  
**\$** the last argument *arg* in a line  
**%** the word *str* matched by the most recent **!*str*?** search  
***m-n*** a range of words from *m* to *n* (*-n* is synonymous with **0-*n***)  
**\*** all *arg*’s  
***m*\*** synonymous with ***m*-\$**  
***m-*** from *arg m* to end, omitting last word

### Event Modifiers

**h** removes a trailing pathname component, leaving the head  
**t** removes all leading pathname components, leaving the tail (filename in path)  
**r** removes a trailing suffix of the form .xxx, leaving the basename

**e** removes all but the trailing suffix  
**p** prints the resulting command but does not execute it  
**q** quotes the substituted words, escaping further substitutions  
**m** quotes the substituted words, breaking them into words at blanks and newlines  
**&** repeats the previous substitution  
**g** causes s/old/new/ or & to be applied over the entire event line  
**G** apply the following ‘s’ modifier just once to each word in the event line.

### Special Commands

**$\wedge s$   $\wedge c$   $\wedge q$**  ..... pause, cancel, resume current process  
**popd** ..... switch to last saved *dir* from index  
**pushd *dir*** .... save *dir* to index  
 **$\wedge X \wedge E$**  ..... work on line under cursor in default EDITOR and execute it afterwards  
 **$\wedge \#$**  ..... prepend hash to command, then jump to next line (... and reedit later)

[github.com/emzap79/QRCs](https://github.com/emzap79/QRCs)

emzap79@gmail.com

This TeXfile is based on Gabriel B. Burcas © git-qrc.tex and has then been modified to my own requirements, with permission!