

The Bash Shell 🚧

Bash shell @ Wikipedia GNU Bash home page	The Bash shell is a POSIX compliant shell used as the Default GNU-Linux. 🚧 This is an early version. • This page is a quick reference to various topics related to the Bash Shell. See PEL specific information here: §1 - UNIX Shell			
Shell Line Editing Keyboard Shortcuts • Using the Emacs mode • readline(3) man page	The keystrokes described below use the Emacs key names like what is used inside all PEL PDF tables. • C-a means holding the control key and hitting the letter a. • M-a means holding the Meta key (which is often mapped to the Alt key) and hitting the letter a. • Note that under Bash, like Emacs, you can type the Esc key, and then hit the a key as an alternative to the M-a key combination. • Both use the GNU readline library.			
Check Bash line edit mode	set -o	Type the set -o command to list all bash features. Two modes are supported: emacs and vi. • Use set -o emacs to activate the Emacs mode. Type help set to get more information about the set command.		
Activate Emacs Line Editing Mode	set -o emacs	Activate Emacs line editing mode.		
Navigation	• In the table: a  means <i>any</i> character, DEL is backspace (delete backwards).			
Begin/end of line	C-a	Move cursor to beginning of line	C-e	Move cursor to end of line
Move by character	C-b	Move cursor backward one character	C-f	Move cursor forward one character
Move by word	M-b	Move cursor backward one word	M-f	Move cursor forward one word
Move to specific char	M-C-] 	Move backward to character 	C-] 	Move forward to character 
Toggle between edge and current char	C-x C-x	Move cursor back to the last position where this command was issued. • If it was never issued for this command move to the beginning of the line.		
Edit Command Text				
Convert letter case	M-l	Lowercase all forward characters	M-u	Uppercase all forward characters
Transpose (exchange)	C-t	Transpose character at cursor and previous one.	M-t	Transpose word at cursor and previous one.
Undo	• C-_ • C-x C-u	Undo last change		
Delete/Kill text	Text deleted cannot be yank'ed back. Text kill can be yank'ed back because it is copied inside the kill ring.			
See § Cut & Paste	 All kill commands issued one after the other fill the <i>same</i> kill buffer entry. Type a non-kill command key to stop filling same entry.			
Delete one chacacter	DEL	Delete backward (delete the character before cursor)	C-d	Delete forward (delete the character at cursor). ⚠ When cursor is at beginning of the line it exits the shell
Kill toward begin/end of word	M-DEL	Kill text from cursor to beginning of current word. • If cursor is at beginning of a word, kill previous word.	M-d	Kill text from cursor to end of current word.
Kill word backward	C-w	Delete word backwards		
Kill toward begin/end of line	C-u	Kill text from cursor to beginning of line.	C-k	Kill text from cursor to end of line.
Yank killed text	Text that was previously killed by the above commands can be yank'ed back at cursor. The kill ring holds many entries of killed text.			
Yank text from kill ring	C-y	Yank text from the kill-ring and insert it at the cursor location.		
Replace last yank from rotated kill ring	M-y	Replace just-yanked text (from C-y) with text from the next position in the kill ring. Type M-y right after C-y and as many time as necessary to insert the text from the required kill ring entry.		
Control terminal				
Clear screen	C-l	clear screen and move cursor to the top-left corner. This is the same as the clear command		
Output flow control	C-s	Stop screen output.	C-q	Allow output to screen (after a previous C-s issued stop)
Command Control	C-c	Terminate the command, do not execute it.	C-z	Suspend/stop the command.
Exit shell	C-d	Exit the shell if cursor is at the beginning of the line.		
Prompt Bash History • See also: dvorka/hstr box-style history searching	Bash support prompt history, it can be enabled or disabled at the ~/.bashrc. To enable it: • Place the line set -o history in ~/.bashrc. • Ensure that the file identified by HISTFILE (which defaults to ~/.bash_history) is owned by your user and group. • If not, you may be able to use sudo chown to change the ownership. Use the history command to see the entire command line history.			
Use last word of last command	M-.	Insert last word of previous command at the prompt.		
To previous/next history entry	C-p	Move to previous history entry.	C-n	Move to next history entry.
To first/last history entry	M-<	Move to beginning (top) of history list.	M->	Move to end (bottom) of history list
Reverse incremental search	C-r	Reverse incremental search. Searches incrementally from the current command line up the history list. • This prints a (reverse-i-search)^ ' : prompt. Type the searched string. Then type: • RET to execute, C-r to search again, C-s to search forward, (see below) • C-c or C-g to quit, C-j to copy into the current command line and allow further editing.		
Forward incremental search	C-s	Forward incremental search. Search incrementally from the current line in the history list. ⚠ If the terminal you use has XON/XOFF flow control (as it's normally the case), this will not work. • To activate C-s search in the terminal, issue the command stty -ixon to disable XON/XOFF software flow control. • stty ixon re-enables it. • To permanently activate it, only in interactive shells, add the following in the ~/.bashrc file: <code>[[\$- == *i*]] && stty -ixon</code>		
Reverse/forward search	M-p	Reverse search, non.incremental.	M-n	Forward search non-incremental.
	These commands search once at the prompt: type the search string and ENTER to execute search.			
Re-execute command from history.	C-o	Execute the current item in the history list and advance to the next one. 🖐 Use it to re-execute a sequence of commands from the history list.		
History expansion	The following commands are executed after RET is typed. Any text can be placed before or after. The expansion occurs when typing RET .			
Run past command	!!	Repeat last command	!<i>number</i>	Repeat history item <i>number</i>
Repeat identified command	!<i>string</i>	Repeat last history <i>starting with string</i>	!<i>?string</i>	Repeat last history list item <i>containing string</i>
Show what would be retrieved	!<i>string</i>:p	Print the command that is retrieved by <i>!<i>string</i></i>		
Retrieve portion of last command	!\$	Last word of the previous command	!*	Repeat the previous command except for the first word.
Show what would be retrieved	!\$:p	Print what !\$ would retrieve.	!*:p	Print what !* would retrieve.
Tab completion	Command line completion can be done in Bash. It can be further specialized via Bash programmable completion .			
Command completion	Tab	Press Tab once to complete sole possibility. • If there more than 1 choice, press Tab again.		
Display possible completions	M-?	Print all possible completions.		
Insert possible completions	M-*	Insert all possible completions at cursor in the current line.		