

grep [OPTION]... PATTERN [FILE]...

-i	case insensitive
-n	line numbers
-r	recursive
-L	only list files that don't match

* can put multiple file names
 * PATTERN is a basic regex by default
 * egrep for extended regex

PyTyle3

Ctrl-Alt-v	tile
Ctrl-Alt-BkSp	untile
Ctrl-Alt-s	⇩ master window space
Ctrl-Alt-r	⇩ master window space
Ctrl-Alt-g	close master
Ctrl-Alt-d	add new master
Ctrl-Alt-c	rotate windows clockwise
Ctrl-Alt-f	float
Ctrl-Alt-q	quit

terminator

Ctrl+Shift+O	Split horizontally
Ctrl+Shift+F	Split vertically
Ctrl+Shift+F	search
(Shift+)Win+	rotate windows
R	(counter)clockwise
Alt+(arrow)	Move terminal focus ...

find [-HLP] [path...] [expression]

-P	never follow symlinks
-L	follow symlinks
[expression]	
* options	-maxdepth #
	-mindepth #
* tests	(+n,-n,n)
	-atime n n*24h
	-amin n
	-executable
	-group <i>name</i> or ID
	-size n[<i>cwbkMG</i>]
	-type [<i>bcdfpls</i>]
	-user <i>name</i> or ID
	-writable
* actions	-delete
	-exec <i>cmd</i>

* path can be specific or .

Referencing previous commands

\$?	exit status of last command
\$0	
\$#	# args passed to current script/ function
\$*	list of args passed, as a space-separated string
@\$	list of args passed, as a delimited list
\$_	most recent arg
\$!	PID of last bg command
\$\$	pid of current shell
\$-	option flags used
\$n	reference to arg <i>n</i> in last command

Bash History: "Event Designators"

! <i>n</i>	Repeat <i>n</i> th command
!- <i>n</i>	Repeat command <i>n</i> lines back
!!	Last command; synonymous w/ !-1
! <i>string</i>	Most recent command starting with <i>string</i>
!? <i>string</i>	Most recent command containing <i>string</i>
?	
^ <i>str1</i> ^ <i>s</i>	repeat last command, replacing <i>str1</i> w/ <i>str2</i>
!!: <i>s/str1/str2</i>	synonymous with the above
!\$	last arg of last cmd; synonymous w/ !:\$.

History "Word Designators"

\$	last argument
0	0 th argument, usually the cmd word (zero)
<i>n</i>	the <i>n</i> th word (<i>n</i> -1 arg)
^	1st argument (word 1)
%	word matched by most recent <i>str</i> ? search
*	all words except the 0 th (i.e., all args)

Word designator is separated from event by a colon ':' unless it is ^,\$,*,-,%. For example, !:\$ becomes !\$. Words are counted from 0, so the command itself is 0. Words can be used with events. For example, !foo:3 finds the last command starting with *foo*, then takes the 3rd word (2nd argument).



History: Modifiers

[g]s/str1/str2

& repeat prev substitution

p prints resulting command but doesn't execute it

e removes all but trailing suffix (e.g., .txt)

r removes trailing suffix, keeps basename

h removes "trailing" last dir in pathname, keeping "head" (path to that directory's directory)

t removes all dirs in path except the last one ("tail")

Used after the event and word designators. Gets very complex. I'm only showing the sed-like ones here.

pacman

-S *pkg* install

-Su upgrade all out-of-date pkgs (prompts for confirmation 1st)

-Sy refresh pkg list

-Ql *pkg* list all files owned by pkg

-Qs search for package (local)
<regex
>

-Ss search repo
<regex
>

-Qi *pkg* pkg info (local)

-Si *pkg* pkg info (repo)

-R *pkg* remove pkg

-Rn don't back up configs
pkg

-Qm list all pkgs not found in DB (manually installed)

-T check all dependencies

-Qc view changelog
pkg

pacman (cont)

-Qd packages installed as dependencies

-Qtd orphan packages

-Qu view all out-of-date pkgs

-Sii info re: *pkg* & what other pkgs depend on it

-Sl list all pkgs in specified repo
[*repo*]

-Suu downgrade pkgs to version in repo (i.e., if installed from testing repo, downgrade to stable)

-Sw download but don't install
pkg

-Syy force refresh

-Syu update + upgrade

-Syu update + upgrade all, then install *pkg* if still not there

-U install from local file (will get /path/ dependencies from repo)

