

```
[18:44 marie@prosoitos:~]$ Fun tools to make  
your life in the command line easier
```

Marie-Hélène Burle

training@westgrid.ca

February 19, 2020

fzf: fuzzy finder

junegunn / fzf

Watch 354 Unstar 26.1k Fork 1k

Code Issues 172 Pull requests 27 Actions Wiki Security Insights

A command-line fuzzy finder

fzf go bash zsh fish vim neovim cli unix

1,688 commits 2 branches 0 packages 86 releases 115 contributors MIT

Branch: master New pull request Create new file Upload files Find file Clone or download

junegunn Use rounded corners Latest commit 577024f 6 days ago

.github	RTFM, please	2 months ago
bin	[fzf-tmux] Avoid unnecessary recovery of window options	2 years ago
doc	Update README-VIM	last month
man/man1	Add preview-fg and preview-bg for --color	25 days ago
plugin	[vim] Encode list source to codepage (#1794)	21 days ago
shell	[bash-completion] Fix default alias/variable completion	17 days ago
src	Use rounded corners	6 days ago
test	Add clear-query and clear-selection	last month
.gitignore	Ignore `zcompile` output files (*.zwc files) (#1775)	last month
.travis.yml	Build with Go 1.13	2 months ago
BUILD.md	Update build instruction	11 months ago
CHANGELOG.md	0.20.0	20 days ago
Dockerfile	[docker] Fix gem install option in Dockerfile	9 months ago
LICENSE	Update license: 2016	4 years ago
Makefile	Support building on machines with `uname -m` == `aarch64` (#1710)	3 months ago
README-VIM.md	Update README-VIM	last month
README.md	[bash-completion] Fix default alias/variable completion	17 days ago
go.mod	Build with Go 1.13	2 months ago

Many commands return long outputs

Get running processes:

```
ps -ef
```

UID	PID	PPID	C	STIME	TTY	TIME	CMD
root	1	0	0	Jan06	?	00:00:16	/sbin/init
root	2	0	0	Jan06	?	00:00:00	[kthreadd]
root	3	2	0	Jan06	?	00:00:00	[rcu_gp]
root	4	2	0	Jan06	?	00:00:00	[rcu_par_gp]
root	6	2	0	Jan06	?	00:00:00	[kworker/0:0H-kblockd]
root	8	2	0	Jan06	?	00:00:00	[mm_percpu_wq]
root	9	2	0	Jan06	?	00:00:00	[ksoftirqd/0]

(Truncated output)

Not user friendly.

Depending on settings, it might not even be possible to scroll up to the line of interest.

Classic tool to make this easier: `less`

```
ps -ef | less
```

UID	PID	PPID	C	STIME	TTY	TIME	CMD
root	1	0	0	Jan06	?	00:00:16	/sbin/init
root	2	0	0	Jan06	?	00:00:00	[kthreadd]
root	3	2	0	Jan06	?	00:00:00	[rcu_gp]
root	4	2	0	Jan06	?	00:00:00	[rcu_par_gp]
root	6	2	0	Jan06	?	00:00:00	[kworker/0:0H-kblockd]
root	8	2	0	Jan06	?	00:00:00	[mm_percpu_wq]
root	9	2	0	Jan06	?	00:00:00	[ksoftirqd/0]

(Truncated output)

- prints output one screen at a time
- search functionality

But `fdf` allows much more:

```
ps -ef | fzf
```

As with many command line utilities, the documentation is in the manual page:

```
man fzf
```


Let's go over an example

I would like to reuse the command `sudo chown marie:marie /media/restic` which is in my command line history.

history

(grumble, grumble...)

history | fzf

Exact match to avoid escaping expressions with `'`

```
history | fzf -e
```

Reverse order...

```
history | fzf -e --tac
```

...but without sorting

```
history | fzf -e --tac +s
```

Let's get rid of the history number

```
history |  
  fzf -e --tac +s |  
  sed -E 's/ *[0-9]* *//'
```

Maybe we could send the result to the clipboard for extra convenience

```
history |  
  fzf -e --tac +s |  
  sed -E 's/ *[0-9]* *//' |  
  xclip -r -selection clipboard
```


We could even create a little function

```
his() {          # search history, send selection to clipboard
  history |
  fzf -e --tac +s |
  sed -E "s/ *[0-9]* *//" |
  xclip -r -selection clipboard
}
```

2nd example

I would like to kill 2 apps: mictray and pasystray

```
ps -ef
```

(grumble, grumble...)

```
ps -ef | less
```

(grumble)

```
ps -ef | fzf
```

We already know the exact match flag

```
ps -ef | fzf -e
```

`-m` to select multiple entries (with `Tab`)

```
ps -ef | fzf -e -m
```

What we want is the pid, which is the 2nd element

```
ps -ef |  
  fzf -e -m |  
  awk '{print $2}'
```


And once we have it, we want to terminate it (with `kill`)

```
ps -ef |  
  fzf -e -m |  
  awk '{print $2}' |  
  xargs kill
```

If this fails, instead of sending the terminate signal (`SIGTERM` or `15` , which is `kill` 's default), we can send the kill signal (`SIGKILL` or `9`)

```
ps -ef |  
  fzf -e -m |  
  awk '{print $2}' |  
  xargs kill -${1:-9}
```

Of course, one can always create a function and add sophistications

```
killlp() {                                     # kill process
    local pid
    if [ "$UID" != "0" ]; then
        pid=$(ps -f -u $UID | sed 1d |
              fzf -i -e -m | awk '{print $2}'); else
        pid=$(ps -ef | sed 1d |
              fzf -i -e -m | awk '{print $2}')
    fi
}
```

(Truncated code. View full code at: https://westgrid-webinars.netlify.app/cli_tools/#!/26)

Last example (getting fancy)

I would like to find a commit and copy its hash.

```
alias glNoGraph='git log --color=always \  
--format="%C(cyan)%h%Creset %C(blue)%ar%Creset%C(auto)%d%Creset \  
%C(yellow)%s%+b %C(black)%ae%Creset" "$@"'  
  
_gitLogLineToHash="echo {} |  
grep -o '[a-f0-9]\{7\}' |  
head -1"
```

(Truncated code. View full code at: https://westgrid-webinars.netlify.app/cli_tools/#!/28)

autojump: smart cd

wtng / autojump

Watch 168 Unstar 10.6k Fork 527

Code Issues 125 Pull requests 35 Actions Projects 0 Wiki Security Insights

A cd command that learns - easily navigate directories from the command line

autojump python filesystem command-line-tool

839 commits 1 branch 0 packages 53 releases 53 contributors View license

Branch: master New pull request Create new file Upload files Find file Clone or download

wtng Merge branch 'lilydjwg_move_temp_file_to_same_device' Latest commit 06e082c on 9 Sep 2018

bin	Bump to 22.5.3.	last year
docs	Bump to 22.5.3.	last year
tests	Add match consecutive tests.	4 years ago
tools	(Running pre-commit) Use Yelp-style indentation.	3 years ago
.gitattributes	(Running pre-commit) Run end of file fixer.	3 years ago
.gitignore	Ignore .pytest_cache/	last year
.pre-commit-config.yaml	Fix flake8 ignore option.	last year
.travis.yml	Add tox for testing with different Python versions.	6 years ago
AUTHORS	update authors list	7 years ago
CHANGES.md	Add minor documentation fixes.	last year
LICENSE	Remove license from individual files, consolidate information into AU...	8 years ago
Makefile	Automatically update manpage header when generating docs.	last year
README.md	Clarify between first class and community supported items.	last year
install.py	(Running pre-commit) Use Yelp-style indentation.	3 years ago
tox.ini	Add match consecutive tests.	4 years ago
uninstall.py	(Running pre-commit) Use Yelp-style indentation.	3 years ago

README.md

ranger: file manager

The screenshot shows the GitHub repository page for 'ranger/ranger'. At the top, it displays the repository name 'ranger / ranger' and various statistics: 85 users, 153 watchers, 6.8k stars, and 492 forks. Below this, there are navigation tabs for Code, Issues (370), Pull requests (51), Actions, Projects (4), Wiki, Security, and Insights. The main heading describes it as 'A VIM-inspired filemanager for the console' with a link to 'https://ranger.github.io/'. There are also tags for 'file-manager', 'vim', 'console', 'file-launcher', and 'file-preview'. A summary bar shows 4,326 commits, 15 branches, 0 packages, 48 releases, 151 contributors, and GPL-3.0 license. Below this, there are buttons for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find file', and 'Clone or download'. The main content area shows a list of files and folders with their commit messages and dates. The most recent commit is by 'toonn' with the message 'Flip the __release__ flag' from 6 days ago. The file list includes folders like '.github', 'doc', 'examples', 'ranger', and 'tests', and files like '.flake8', '.gitignore', '.pylintrc', '.travis.yml', 'AUTHORS', 'CHANGELOG.md', 'CONTRIBUTING.md', 'Dockerfile', 'HACKING.md', 'LICENSE', 'Makefile', 'README.md', and 'ranger.py'.

ranger / ranger

Used by 85 Watch 153 Unstar 6.8k Fork 492

Code Issues 370 Pull requests 51 Actions Projects 4 Wiki Security Insights

A VIM-inspired filemanager for the console <https://ranger.github.io/>

file-manager vim console file-launcher file-preview

4,326 commits 15 branches 0 packages 48 releases 151 contributors GPL-3.0

Branch: master New pull request Create new file Upload files Find file Clone or download

toonn Flip the __release__ flag Latest commit 3965e3e 6 days ago

.github	Seperate doctests and pytests	3 months ago
doc	Tax loss harvesting release, nothing to see here	6 days ago
examples	Merge branch 'dmitmel-examples-bash-automatic-cd'	6 days ago
ranger	Flip the __release__ flag	6 days ago
tests	Refactor the bookmark validation code	11 months ago
.flake8	Add ignore for 'W503' to '.flake8' to follow new pep8	2 years ago
.gitignore	Revert "Merge branch 'slackhead-venv-gitignore'"	10 months ago
.pylintrc	Pylint: Disable the stop-iteration-return check	2 years ago
.travis.yml	Replace pipenv with a simple requirements.txt	last year
AUTHORS	Patched patch of intel microcode patch	2 years ago
CHANGELOG.md	Tax loss harvesting release, nothing to see here	6 days ago
CONTRIBUTING.md	update CONTRIBUTING.md	2 years ago
Dockerfile	Add docker entrypoint	2 years ago
HACKING.md	Fix spelling in Patching section	6 months ago
LICENSE	Move license from 'AUTHORS' to 'LICENSE'	3 years ago
Makefile	Tax loss harvesting release, nothing to see here	6 days ago
README.md	Tax loss harvesting release, nothing to see here	6 days ago
ranger.py	Apply requested changes to ranger.py and enforce full POSIX compliance	2 months ago

Here is a little trick with `fzf` for those with short memory (like me) to get reminded of the keybindings

```
alias kr='grep -E "^map\ " $HOME/.config/ranger/rc.conf |  
fzf -i -e +s'
```


Alternatives to ranger

Written in C: `nnn`

In Go: `lf` (that's lower case `L`)

And in bash: `fff` (you will have to visit the page to know what the 3 `f` stand for. I'll just say here that it is a *Very Fast File-manager*)

If you would like to explore other fun utilities, here is a little selection:

- `tig` . If you read it backward, you will get some hint as to what it is about
- `rg` . Stands for “RIP grep”...
- `ag` . Similar to `rg` . You need to have some background in Latin or in chemistry to know right away that it is the *silver* searcher
- `fd` . If you have always found `find` ’s syntax awkward, this might be for you. And it is fast. *Very fast*
- `fasd` . Similar to autojump, but also works on files. However the project is currently not being developed anymore

```
3:07 marie@prosoitos: ~$ Questions?
```