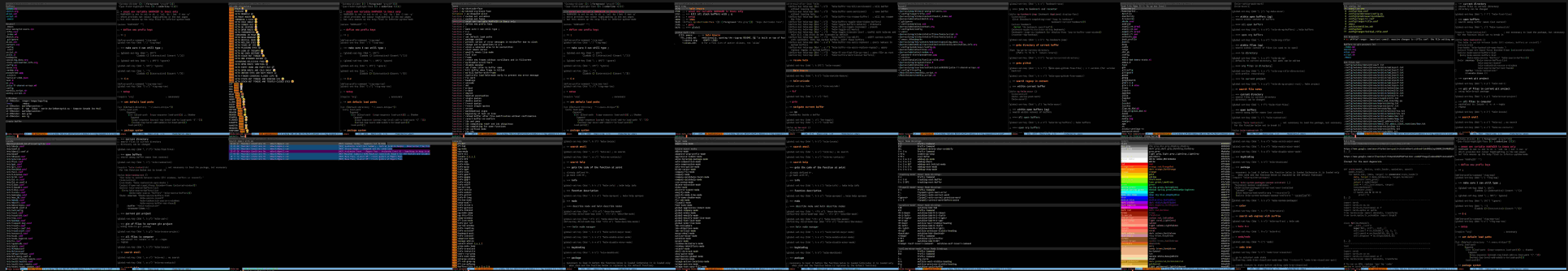


# A few of our favourite tools



**Marie-Hélène Burle & Alex Razoumov**

**training@westgrid.ca**

**March 16, 2022**

# lazygit

(Terminal UI for Git written in Go)

```
status
10:0 marie ~ m

Files - Submodules
?? .home/marie/pazvus/prog/mcluu/main.tf -undo-tree- .gz
?? .home/marie/pazvus/prog/remoteconfig/.bashrc -undo-tree- .gz
?? .home/marie/pazvus/prog/remoteconfig/.gitignore -undo-tree- .gz
?? .home/marie/pazvus/prog/tcl/content/webinars/tools.org -undo-tree- .gz
?? .home/marie/pazvus/prog/tcl/content/webinars/lupscaling_data.org -undo-tree- .gz
?? .home/marie/spherical001.nc -undo-tree- .gz
?? .home/marie/system_log/mail_sync.log -undo-tree- .gz
?? .home/marie/system_log/monthly.log -undo-tree- .gz
?? .home/marie/system_log/weekly.log -undo-tree- .gz

zsh/custom
M alias.zsh
M .aspell.en.pws
M .emacs
M .gitconfig
?? default_jupyter_kbd.json

Local Branches - Remotes - Tags
* m 10:0

Commits - Reflog
1 of 1
3da3fb17 MB alias: add comment
1b091ba9 MB Add xautolock on/off to ssaf
1e30b5e MB alias: update pdf
1a226fd9 MB remove new line at end of .gitignore as that make it play well with
744bf634 MB update gitignore
9ce01f1e MB alias: format + minor comment edit
e4525247 MB alias: update path
b01dc9b MB alias: better alias for llpp (pdf) kbd
46c934e3 MB add alias for ebook kbd
eb38834b MB alias: big revamp of all clean functions
bcf548c8 MB alias: add alex function to send large file over the net
61071ef1 MB alias: fix wacom tablet code
74fdabd2 MB ranger: make all fzf commands to cycle
36c77004 MB xinitrc: better comments

Stash
0 of 0

Unstaged Changes
modified: .emacs

@ .emacs:146 @ mouse-3: menu for this date)
'(company-frontends
'(company-pseudo-tooltip-unless-just-one-frontend company-echo-metadata-frontend company-preview-if-just-one-frontend))
'(company-idle-delay nil)
'(company-insertion-triggers nil)
'(company-lighter-base "")
'(company-minimum-prefix-length 10)
'(company-quickhelp-color-background "brown4")
@ .emacs:232 @ mouse-3: menu for this date)
'(emojify-download-emojis-p t)
'(emojify-emoji-set "tweemoji-v2-22")
'(emojify-emoji-styles '(unicode))
'(emojify-inhibit-in-buffer-functions nil)
'(erc-autojoin-channels-alist '(("localhost")))
'(erc-disable-ctcp-replies t)
'(erc-header-line-face-method t)
@ .emacs:2838 @ $0` (yas-escape-text yas-selected-text)")

(require 'package) ; necessary

;; (setq package-archives '(("gnu" . "https://elpa.gnu.org/packages/")
;; ("melpa" . "https://melpa.org/packages/")
;; ("org" . "https://elpa.nongnu.org/nongnu/")))

(add-to-list 'package-archives '("melpa" . "https://melpa.org/packages/") t)

(package-initialize)
@ .emacs:2849 @ $0` (yas-escape-text yas-selected-text)")
(let ((inhibit-message t))
  (message "curl: (6) Could not resolve host: emoji.slack-edge.com"))

(let ((inhibit-message t))
  (message "curl: (6) Could not resolve host: secure.gravatar.com"))

(let ((inhibit-message t))
  (message "curl: (6) Could not resolve host: avatars.slack-edge.com"))

(let ((inhibit-message t))
  (message "Download Failed. STATUS: exit, EVENT: exited abnormally with code 6, URL: https://avatars.slack-edge.com/2021-01-11/1621414413302_7290da985afe09d5d61_32.jpg, NAME: /tmp/57ef60fe94e4862ca8a9f8b8402787b1.jpg, OUTPUT:"))

(let ((inhibit-message t))
  (message "[error] [Compute Canada] REQUEST FAILED. URL: \"https://slack.com/api/rtm.connect\", PARAMS: ((\"mpim_aware\" . \"1\") (\"presence_sub\" . \"true\")) , ERROR-THROWN: (error \"exited abnormally with code 6\"), SYMBOL-STATUS: parse-error, DATA: nil"))

;; ** change yes-no questions to y-n

(defalias 'yes-or-no-p 'y-or-n-p)
@ .emacs:3302 @ displayed anywhere else."
(bookmark--jump-via bookmark (or display-func 'pop-to-buffer-same-window))
(recenter-top-bottom))

1-5: jump to panel, [/]: scroll left/right, ctrl+g: cancel, ctrl+k/ctrl+j: scroll, q: quit, x: menu, ◀▶▲▼: navigate
Donate Ask Question 0.33
```

**lazygit** brings the ease of use of a visual Git tool to the command line

Unlike slow, buggy, & limited Git GUIs, it is nimble, portable, & super fast

In the following demo, as an additional visual goody, I will introduce **diff-so-fancy** to display the outputs of `git diff`

# Bat

(“A cat clone with wings”)

File: /home/marie/parvus/prog/tdoc/djl/cjl/parallel/generic\_loop/both.jl

```
1 #####
2 # sequential
3
4 # run it with:
5 # julia parallel.jl
6 #####
7
8 using BenchmarkTools
9
10 function slow(n::Int, digits::String)
11     total = 0.
12     for i in 1:n
13         if !occursin(digits, string(i))
14             total += 1. / i
15         end
16     end
17     println("total = ", total)
18 end
19
20 @btime slow(Int64(1e9), "9")
21
```

**bat** is a great tool to display the content of files in your terminal

The standard Unix utility for this task is **cat**, but bat adds a lot to it

# Syntax highlighting...

...in a huge number of languages. See the impressive list with:

```
bat -L
```

```
cat /home/marie/parvus/prog/slides/static/files/tools/julia.jl
```

# Syntax highlighting...

...in a huge number of languages. See the impressive list with:

```
bat -L
```

```
cat /home/marie/parvus/prog/slides/static/files/tools/julia.jl
```

```
#####  
# sequential  
  
# run it with:  
# julia parallel.jl  
#####  
  
using BenchmarkTools  
  
function slow(n::Int, digits::String)  
    total = 0.  
    for i in 1:n  
        if !occursin(digits, string(i))  
            total += 1. / i  
        end  
    end  
    println("total = ", total)  
end  
  
@btime slow(Int64(1e9), "9")
```

# Syntax highlighting...

...in a huge number of languages. See the impressive list with:

```
bat -L
```

```
bat /home/marie/parvus/prog/slides/static/files/tools/julia.jl
```

```
File: /home/marie/parvus/prog/slides/static/files/tools/julia.jl
Size: 472 B
1 #####
2 # sequential
3
4 # run it with:
5 # julia parallel.jl
6 #####
7
8 using BenchmarkTools
9
10 function slow(n::Int, digits::String)
11     total = 0.
12     for i in 1:n
13         if !occursin(digits, string(i))
14             total += 1. / i
15         end
16     end
17     println("total = ", total)
18 end
19
20 @btime slow(Int64(1e9), "9")
```



# Smart behaviour with pager

Uses **less** by default with options `-RFX`

Very convenient: quits the pager automatically if the entire content fits on one script

# Integration with Git

```
File: /home/marie/parvus/prog/slides/static/files/tools/two_layer_net_tensor.py
Size: 2.3 KB

1 | import torch
2 | from torchvision import datasets, transforms
3 | from matplotlib import pyplot as plt
4 |
5 | # To run on GPU, replace 'cpu' by 'cuda'
6 | device = torch.device('cpu')
7 | dtype = torch.float
8 | epochs = 30
9 |
10 | # * Prepare data
11 |
12 | train = datasets.MNIST(
13 | ~   # './data',
14 | ~   '~/projects/def-sponsor00/data',
15 | ~   train = False,
16 | ~   download = False,
17 |   transform = transforms.Compose([
18 |     transforms.ToTensor(),
19 |     transforms.Normalize((0.1307,), (0.3081,))])
20 |
21 | test = datasets.MNIST(
22 | ~   # './data',
23 | ~   '~/projects/def-sponsor00/data',
24 | ~   train = False,
25 | ~   download = False,
26 |   transform = transforms.Compose([
27 |     transforms.ToTensor(),
28 |     transforms.Normalize((0.1307,), (0.3081,))])
29 |
30 | N, D_in, H, D_out = 60000, 784, 100, 10
31 |
32 | x = train.data.view(60000, 784).type(dtype)
33 | y = train.targets.unsqueeze(1).expand(60000, 10).type(dtype)
34 |
35 | # Randomly initialize weights
36 | w1 = torch.randn(D_in, H, device = device, dtype = dtype)
37 | w2 = torch.randn(H, D_out, device = device, dtype = dtype)
38 |
39 | + optimizer = torch.optim.Adam(model.parameters(), lr=learning_rate)
40 | + for t in range(500):
41 | +   # Forward pass: compute predicted y by passing x to the model.
42 | +   y_pred = model(x)
```

# Git diff

bat -d

```
File: /home/marie/parvus/prog/slides/static/files/tools/two_layer_net_tensor.py
Size: 2.3 KB

11
12 train = datasets.MNIST(
13 ~     # './data',
14 ~     '~/projects/def-sponsor00/data',
15 ~     train = False,
16 ~     download = False,
17     transform = transforms.Compose([
18         transforms.ToTensor(),
19     ])
20 ~
21 test = datasets.MNIST(
22 ~     # './data',
23 ~     '~/projects/def-sponsor00/data',
24 ~     train = False,
25 ~     download = False,
26 ~
27     transforms.ToTensor(),
28     transforms.Normalize((0.1307,), (0.3081,)))
29 ~
30 N, D_in, H, D_out = 60000, 784, 100, 10
31 ~
32 ~
33 ~
34 ~
35 ~
36 ~
37 w2 = torch.randn(H, D_out, device = device, dtype = dtype)
38 ~
39 + optimizer = torch.optim.Adam(model.parameters(), lr=learning_rate)
40 + for t in range(500):
41 +     # Forward pass: compute predicted y by passing x to the model.
42 +     y_pred = model(x)
43 +
44 +     # Compute and print loss.
45 +     loss = loss_fn(y_pred, y)
46 +     print(t, loss.item())
47 +
48 +     writer.add_scalar("training loss", loss, t)
49 +     # Before the backward pass, use the optimizer object to zero all of the
50 +     # gradients for the Tensors it will update (which are the learnable weights
51 +     # of the model)
52 +     optimizer.zero_grad()
53 +
54 +     # Backward pass: compute gradient of the loss with respect to model parameters
55 +     loss.backward()
56 +
```

# Man page syntax highlighting

Even colourizes your man pages if you set the `MANPAGER` environment variable (e.g. in your `.bashrc`):

```
export MANPAGER="sh -c 'col -bx | bat -l man -p'"
```

# Useful flags

```
bat -n          # Remove frame
```

```
bat -p          # Remove frame & line numbers
```

I personally have an alias for this: `alias rat='bat -p'`

```
bat -pp        # Remove frame & line numbers & don't use pager
```

```
bat -f         # Keep colour highlighting after piping
```

I made this the default behaviour for `bat` with the alias: `alias bat='bat -f'`

# Fully customizable

The defaults are great, but if they don't suit you, you can change tab width, language, wrapping method...

```
bat --config-file          # Location of config file (if it exists)
```

```
bat --generate-config-file # Create config file template
```

```
bat --list-themes         # List themes with examples
```

# ripgrep

(A fast recursive alternative to grep)

```
13:57 m 1 ~/parvus/prog/slides rg tools
content/git_branches/_index.md
415:Many GUI applications for Git as well as powerful text editors such as Emacs and Vim offer merge tools

content/r_gis/_index.md
103:`basemaps` is not on CRAN and needs to be installed from GitHub thanks to `devtools`:
106:install.packages("devtools")
107:devtools::install_github("16EAGLE/basemaps")

content/r_gis_rc/_index.md
98:`basemaps` is not on CRAN and needs to be installed from GitHub thanks to `devtools`:
101:install.packages("devtools")
102:devtools::install_github("16EAGLE/basemaps")

content/help/_index.md
94:Take the time to look for the official documentation & other high quality sources for the languages & tools you use.

content/cli_tools/_index.md
2:title = "cli_tools"
7:href = "https://westgrid-cli.netlify.app/webinars/cli_tools/"
24:![ ] (/img/cli_tools/cli_title.png)
38:![logo] (/img/cli_tools/fzf.png) (https://github.com/junegunn/fzf)
510:![logo] (/img/cli_tools/autojump.png) (https://github.com/wting/autojump)
518:![logo] (/img/cli_tools/ranger.png) (https://github.com/ranger/ranger)
558:![ ] (/img/cli_tools/questions.png)

content/tools/_index.md
2:title = "tools"
7:href = "https://westgrid-cli.netlify.app/webinars/tools/"
14:## <div style="line-height: 5rem; font-size: 4.2rem; color: black; text-shadow: 0 0 0.7rem white;">A few of our favourite tools</div>
35:{{}}
44:{{}}
66:cat /home/marie/parvus/prog/slides/static/files/tools/julia.jl
68:{{}}
85:bat /home/marie/parvus/prog/slides/static/files/tools/julia.jl
87:{{}}
103:{{}}
114:{{}}
199:{{}}
207:{{}}
215:{{}}
223:{{}}
231:{{}}
239:{{}}
247:{{}}
255:{{}}
263:{{}}
```

**ripgrep** provides the utility `rg`:  
an alternative to `grep` very fast for recursive searches



# Syntax

The syntax is similar to that of `grep`

Because, by default, `rg` is recursive (`grep` isn't) & ignores some files, `rg -uuu` is equivalent to `grep -r`

# Benchmarks

I will compare `grep` and `rg` to recursively search the word “Unix” in my directory “~/parvus/prog/tcl”

```
grep -r Unix ~/parvus/prog/tcl
rg Unix ~/parvus/prog/tcl
```

I will do a cold test (clearing my computer cache before each run) using `hyperfine`

```
hyperfine -i --prepare 'sync; echo 3 | sudo tee /proc/sys/vm/drop_caches' \  
  'grep -r Unix ~/parvus/prog/tcl' \  
  'rg Unix ~/parvus/prog/tcl'
```

# Results

Benchmark 1: `grep -r Unix ~/parvus/prog/tcl`

Time (mean  $\pm$   $\sigma$ ): 1.027 s  $\pm$  0.016 s [User: 0.269 s, System: 0.222 s]

Range (min ... max): 1.007 s ... 1.052 s 10 runs

Benchmark 2: `rg Unix ~/parvus/prog/tcl`

Time (mean  $\pm$   $\sigma$ ): 76.0 ms  $\pm$  5.6 ms [User: 21.5 ms, System: 29.6 ms]

Range (min ... max): 63.0 ms ... 82.4 ms 10 runs

## Summary

'`rg Unix ~/parvus/prog/tcl`' ran

13.51  $\pm$  1.02 times faster than '`grep -r Unix ~/parvus/prog/tcl`'

# What makes rg fast?

Uses parallelism

Uses Rust regex engine with finite automata, SIMD, & aggressive literal optimizations (*regular expressions follow the syntax of `grep -E`; see [this](#) & [this](#) for details*)

By default, ignores files in `.gitignore` & similar, hidden files/directories, & binary files (this behaviour can be changed with flags)

# fd

(Fast & friendly alternative to find)

```
08:38 m ~/parvus/prog/tdoc fd -e jl
dcl/ccl/getting_help/script.jl
dcl/ccl/getting_help/script_fixed.jl
djl/cjl/covid/covid.jl
djl/cjl/covid/covid_old.jl
djl/cjl/dagger/benchmark.jl
djl/cjl/dagger/dagger_oom.jl
djl/cjl/dagger/dagger_threads.jl
djl/cjl/flux/flux.jl
djl/cjl/intro/distributed.jl
djl/cjl/intro/intro.jl
djl/cjl/intro/live_code.jl
djl/cjl/intro/pi.jl
djl/cjl/intro/pi_parallel.jl
djl/cjl/intro/plots.jl
djl/cjl/intro/plots_bkup.jl
djl/cjl/intro/psort.jl
djl/cjl/intro/sort.jl
djl/cjl/intro/variables_types.jl
djl/cjl/jupyter/covid_jupyterdays.jl
djl/cjl/jupyter/hello.jl
djl/cjl/jupyter/literate_cleaned.jl
djl/cjl/jupyter/literate_example.jl
djl/cjl/jupyter/nbinclude.jl
```

**fd** has an intuitive syntax & is **super fast**

# pass

(A command line password manager for Unix systems)

```
15:24 m ~ pass
Password Store
├─ newsite.ca
│  └─ your@email
├─ othersite.com
│  └─ userid
└─ somesite.ca
   └─ your@email
```

`pass` is a command line password manager using **GnuPG** for encryption

`pass` has a Git integration & several extensions

`pass` is easy to use in scripts or combined with other utilities



# Main pass commands

```
pass init your-gpg-id # Initialize password store
pass insert somesite.ca/your@email # Enter password for a site
pass somesite.ca/your@email # Print password
pass -c somesite.ca/your@email # Copy password to clipboard
pass edit somesite.ca/your@email # Edit password
pass generate othersite.com/userid 12 # Generate password of size 12
pass generate -c othersite.com/userid 12 # Same, but copy to clipboard
pass generate -n othersite.com/userid 12 # Same, without special characters
pass rm othersite.com/userid # Delete password
pass # Print content of password store
```

# TRAMP

(“A remote file editing package for Emacs”)

The **TRAMP** (Transparent Remote Access, Multiple Protocol) Emacs package allows to edit remote files as if they were local

You can thus use your init file & usual packages to write files on a remote machine or for another user (e.g. root)

You can also use the Emacs GUI to edit remote files without efficiency loss

It is a convenient alternative to using Emacs on a remote machine after you have `ssh`ed into it or to using `sudo emacs` to edit root files

# Using TRAMP from within Emacs

## Remote file

C-x C-f /ssh:user@server:/home/user/path/to/file RET

## Root

C-x C-f /sudo::/path/to/file RET

# Using TRAMP from the command line

## Remote file

```
emacs /ssh:user@server:/home/user/path/to/file
```

## Root

```
emacs /sudo:./path/to/file
```

# Helm

“Incremental completion & selection narrowing framework”

```
C-j: Go to Line (keeping session)
HO [.emacs]
18: '(TeX-auto-global "-/.emacs.d/auctex/")
316: '(global-aggressive-indent-mode t)
317: '(global-emojify-mode t)
318: '(global-emojify-mode-line-mode t)
319: '(global-git-commit-mode t)
320: '(global-undo-tree-mode t)
321: '(global-visual-line-mode nil)
692: '(org-cycle-global-at-bob t)
2181: '(outshine-org-style-global-cycling-at-bob-p t)
2260: '(smartparens-global-mode t)
2261: '(smartparens-global-strict-mode nil)
2335: '(writeroom-global-effects
2338: '(yas-global-mode t)
2800: (global-set-key (kbd ";") 'ring-map)
2804:;; (global-set-key (kbd "; SPC")
2807:;; (global-set-key (kbd "; SPC") 'ignore)
2809: (global-set-key (kbd "; SPC") 'ignore)
2811: (global-set-key (kbd ";")
2817: (global-set-key (kbd "C-z") 'ring-map-two)
2877:;; useful only when global-emojify-mode is nil
2934: (global-set-key (kbd "<kp-next>") #'fix-word-downcase) ; F+l (l=lower case)
2935: (global-set-key (kbd "<kp-left>") #'fix-word-upcase) ; F+u (u=upper case)
2936: (global-set-key (kbd "<kp-down>") #'fix-word-capitalize) ; F+k (k=capitalize)
2940: (global-set-key (kbd "; h 1") "*" ")
2942: (global-set-key (kbd "; h 2") "**** ")
2944: (global-set-key (kbd "; h 3") "***** ")
2946: (global-set-key (kbd "; h 4") "***** ")
2948: (global-set-key (kbd "; h 5") "***** ")
2950: (global-set-key (kbd "; h 6") "***** ")
2952: (global-set-key (kbd "; h 7") "***** ")
2954: (global-set-key (kbd "; h 8") "***** ")
2963: (global-set-key (kbd "<f12> ." ) "o")
2964: (global-set-key (kbd "<f12> o" ) "o")
2965: (global-set-key (kbd "<f12> 0" ) "o")
2969: (global-set-key (kbd "<f12> -" ) "-" )
2973: (global-set-key (kbd "<f12> _" ) "-" )
2977: (global-set-key (kbd "<f12> d" ) "o")
2981: (global-set-key (kbd "<f12> !" ) "i")
2982: (global-set-key (kbd "<f12> ?" ) "i")
2986: (global-set-key (kbd "<f12> ' o" ) "'")
2987: (global-set-key (kbd "<f12> ' c" ) "'")
2991: (global-set-key (kbd "<f12> \\" o" ) "\\")
2992: (global-set-key (kbd "<f12> \\" c" ) "\\")
2996: (global-set-key (kbd "<f12> \< \< o" ) "\< ")
2997: (global-set-key (kbd "<f12> \> \> c" ) "\> ")
3001: (global-set-key (kbd "<f12> \< o" ) "\< ")
3002: (global-set-key (kbd "<f12> \> c" ) "\> ")
3006: (global-set-key (kbd "<f12> a r" ) "r")
3007: (global-set-key (kbd "<f12> a l" ) "l")
helm occurs L14 (HF) 1331 candidate(s) C-h m:Help TAB:Act RET/f1/f2/f-n:Nhact C-l:Log.suspend C-h c:
pattern: global

(vertical-border (((type w32 tty)) (:inherit mode-line-inactive :foreground
"firebrick4"))))
(vimish-fold-mouse-face ((t (:background "gray15" :foreground "indianred"))))
(vimish-fold-overlay ((t (:background "MistyRose4" :foreground "black"))))
(warning ((t (:foreground "red2" :weight bold))))
(whitespace-empty ((t (:background "light green" :foreground "gray10"))))
(whitespace-hspace ((t (:foreground "orange red"))))
(whitespace-line ((t (:background "gray15"))))
(whitespace-space ((t (:foreground "olivedrab"))))
(whitespace-trailing ((t (:background "brown2"))))
(window-divider ((t (:foreground "gray25"))))
(yas-field-highlight-face ((t (:underline t))))

; * unset env variable MANPAGER in Emacs only
; MANPAGER is set in zsh to sh -c 'col -bx | bat -l man -p'
; which provides bat colour highlighting in the man pages
; but this messes up the help files in inferior-python-mode

(setenv "MANPAGER" "")

; * define new prefix keys

; ** ;

(define-prefix-command 'ring-map)
global-set-key (kbd ";") 'ring-map

; *** make sure I can still type ;

; (global-set-key (kbd "; SPC")
; (lambda () (interactive) (insert ";")))

; (global-set-key (kbd "; SPC") 'ignore)

(global-set-key (kbd "; SPC") 'ignore)

(global-set-key (kbd ";")
(lambda () (interactive) (insert ";")))

; ** C-z

(define-prefix-command 'ring-map-two)
(global-set-key (kbd "C-z") 'ring-map-two)

; * setup

(require 'org) ; necessary

; ** set default load paths

(+ (default-directory) "/.emacs.d/elp/"))
```

The **Helm** package provides an incremental completion & narrowing selection API for search applications in Emacs

Virtually every Emacs search application has built-in integration with Helm

# Grep equivalent

```
C-j: Go to Line (keeping session)
HO [ ,emacs]
18: '(TeX-auto-global "~/emacs.d/auctex/")
316: '(global-aggressive-indent-mode t)
317: '(global-emoji-mode t)
318: '(global-emoji-mode-line-mode t)
319: '(global-git-commit-mode t)
320: '(global-undo-tree-mode t)
321: '(global-visual-line-mode nil)
692: '(org-cycle-global-at-bob t)
2181: '(outshine-org-style-global-cycling-at-bob-p t)
2260: '(smartparens-global-mode t)
2261: '(smartparens-global-strict-mode nil)
2335: '(writeroom-global-effects
2338: '(yas-global-mode t)
2800: (global-set-key (kbd ";") 'ring-map)
2804:;; (global-set-key (kbd " "; SPC)
2807:;; (global-set-key (kbd " "; SPC) 'ignore)
2809: (global-set-key (kbd " "; SPC) 'ignore)
2811: (global-set-key (kbd " "; ))
2817: (global-set-key (kbd "C-z") 'ring-map-two)
2877:;; useful only when global-emoji-mode is nil
2934: (global-set-key (kbd "<kp-next>") #'fix-word-downcase) ; F+l (l=lower case)
2935: (global-set-key (kbd "<kp-left>") #'fix-word-upcase) ; F+u (u=upper case)
2936: (global-set-key (kbd "<kp-down>") #'fix-word-capitalize) ; F+k (k=capitalize)
2940: (global-set-key (kbd " "; h 1) "*" ")
2942: (global-set-key (kbd " "; h 2) "*** ")
2944: (global-set-key (kbd " "; h 3) "**** ")
2946: (global-set-key (kbd " "; h 4) "***** ")
2948: (global-set-key (kbd " "; h 5) "***** ")
2950: (global-set-key (kbd " "; h 6) "***** ")
2952: (global-set-key (kbd " "; h 7) "***** ")
2954: (global-set-key (kbd " "; h 8) "***** ")
2963: (global-set-key (kbd "<f12> ." ) ".")
2964: (global-set-key (kbd "<f12> o" ) "o")
2965: (global-set-key (kbd "<f12> 0" ) "o")
2969: (global-set-key (kbd "<f12> -" ) "-" )
2973: (global-set-key (kbd "<f12> _" ) "_" )
2977: (global-set-key (kbd "<f12> d" ) "d")
2981: (global-set-key (kbd "<f12> !" ) "!")
2982: (global-set-key (kbd "<f12> ?" ) "?")
2986: (global-set-key (kbd "<f12> ' o" ) "'o")
2987: (global-set-key (kbd "<f12> ' c" ) "'c")
2991: (global-set-key (kbd "<f12> \v o" ) "\v o")
2992: (global-set-key (kbd "<f12> \v c" ) "\v c")
2996: (global-set-key (kbd "<f12> \< \< o" ) "\< \< o")
2997: (global-set-key (kbd "<f12> \> \> c" ) "\> \> c")
3001: (global-set-key (kbd "<f12> \< o" ) "\< o")
3002: (global-set-key (kbd "<f12> \> c" ) "\> c")
3006: (global-set-key (kbd "<f12> a r" ) "a r")
3007: (global-set-key (kbd "<f12> a l" ) "a l")
(global-set-key (kbd "C-h") 'help)
(global-set-key (kbd "C-!") 'toggle-suspend)
(global-set-key (kbd "C-h c") 'emacs)

(vertical-border (((type w32 tty)) (:inherit mode-line-inactive :foreground
"firebrick4"))))
(vimish-fold-mouse-face ((t (:background "gray15" :foreground "indianred"))))
(vimish-fold-overlay ((t (:background "MistyRose4" :foreground "black"))))
(warning ((t (:foreground "red2" :weight bold))))
(whitespace-empty ((t (:background "light green" :foreground "gray10"))))
(whitespace-hspace ((t (:foreground "orange red"))))
(whitespace-line ((t (:background "gray15"))))
(whitespace-space ((t (:foreground "olivedrab"))))
(whitespace-trailing ((t (:background "brown2"))))
(window-divider ((t (:foreground "gray25"))))
(yas-field-highlight-face ((t (:underline t))))

;; * unset env variable MANPAGER in Emacs only
;; MANPAGER is set in zsh to sh -c 'col -bx | bat -l man -p'
;; which provides bat colour highlighting in the man pages
;; but this messes up the help files in inferior-python-mode

(setenv "MANPAGER" "")

;; * define new prefix keys
;; ** ;

(define-prefix-command 'ring-map)
(global-set-key (kbd ";") 'ring-map)

;; *** make sure I can still type ;
;; (global-set-key (kbd " "; SPC)
;; (lambda () (interactive) (insert ";")))

;; (global-set-key (kbd " "; SPC) 'ignore)

(global-set-key (kbd " "; SPC) 'ignore)

(global-set-key (kbd " "; ))
(lambda () (interactive) (insert ";")))

;; ** C-z

(define-prefix-command 'ring-map-two)
(global-set-key (kbd "C-z") 'ring-map-two)

;; * setup

(require 'org) ; necessary

;; ** set default load paths

(1+ (default-directory "~/emacs.d/elpa/"))
/home/marie/emacs 61% 2800 Emacs-Lisp s U: /home/marie/emacs
```



# Navigate mark ring

```
C-j: Show this line (keeping session)
mark-ring
3173: ;; ** helm kmacro
2788: ;; * unset env variable MANPAGER in Emacs only
5781: ;; *** kill all slack buffers with ; q
2819: ;; * setup
3060: ;; ** smex
5449: (defface my-shortcode-face '((t (:foreground "dim gray"))) "Hugo shortcodes face")
3185: ;; * goto
5473: ;; **** global

global-mark-ring
3173:.emacs: ;; ** helm kmacro
176:_index.md: Additionally, quoting the ripgrep README, rg "is built on top of Rus
1:sketch.org: <EMPTY LINE>
1:alias.zsh: # For a full list of active aliases, run `alias`

(with-eval-after-load 'helm
  (define-key helm-map (kbd "; k") 'helm-buffer-run-kill-persistent) ; kill buffer
    without leaving helm
  (define-key helm-map (kbd "; s") 'helm-buffer-save-persistent) ; save buffer
    without leaving helm
  (define-key helm-map (kbd "; n") 'helm-buffer-run-rename-buffer) ; will not rename
    the file
  (define-key helm-map (kbd "; h") 'helm-buffers-toggle-show-hidden-buffers)
  (define-key helm-map (kbd "; d") 'helm-toggle-buffers-details) ; d=details
  (define-key helm-map (kbd "; p") 'helm-ff-run-toggle-basename) ; p=path
  (define-key helm-map (kbd "; a") 'helm-toggle-all-marks)
  (define-key helm-map (kbd "; t") 'helm-toggle-truncate-line) ; useful with helm-mu and
    helm kill ring which do not truncate by default
  (define-key helm-map (kbd "; =") 'helm-buffer-run-ediff) ; ediff current buffer
    with candidate. With two marked buffers, ediff those buffers
  (define-key helm-map (kbd "; m m") 'helm-buffer-run-query-replace) ; query replace in
    marked buffers
  (define-key helm-map (kbd "; m r") 'helm-buffer-run-query-replace-regexp) ; query
    replace regexp in marked buffers
  (define-key helm-map (kbd "; r") 'helm-ff-run-find-file-as-root) ; open file as root
  (define-key helm-map (kbd "; i") 'helm-buffers-mark-similar-buffers))

;; ** resume helm

(global-set-key (kbd "; h SPC") 'helm-resume)

;; ** helm kmacro

(global-set-key (kbd "; h k") 'helm-execute-kmacro)

;; ** helm-unicode

(global-set-key (kbd "; h u") 'helm-unicode)

;; * fzf

(global-set-key (kbd "; f s") 'fzf)

;; * goto

;; ** navigate current buffer

;; *** bm
;; bookmarks inside a buffer

(global-set-key (kbd "; u") 'bm-toggle)
(global-set-key (kbd "; e r") 'helm-bm)

(defun my-bm-next ()
  (interactive)
  (helm-ediff-synch-mark))

#helm mark ring* L1 (HF) [8 Candidate(s)] C-h m:HelP TAB:Act RET/f1/f2/f-n:NthAct C-!:Tog.suspend C-!-
pattern: | emacs 64% 3173 Emacs-Lisp s U: /home/marie/.emacs
```

# Navigate kill ring

```
C-j: DoNothing (keeping session)
Kill Ring
go to repo /home/marie/parvus/ptmp/git_workshop_collab
-----
https://www.google.com/search?q=button+quail+chicks&tbn=isch&ved=2ahUKewjvptG06Mj2AhWbBDQI
[...]
https://www.google.com/url?sa=i&url=https%3A%2F%2Ftwitter.com%2Fthegallowboob%2Fstatus%2F1
-----
(Except for the most degenerate
-----
def train(model, device, train_loader, optimizer, epoch):
    model.train()
    for batch_idx, (data, target) in enumerate(train_loader):
        data, target = data.to(device), target.to(device)
        optimizer.zero_grad()
        output = model(data)
        loss = F.nll_loss(output, target)
        loss.backward()
        optimizer.step()
        if batch_idx % 10 == 0:
            print('Tr
[...]
import torch
import torch.nn as nn
import torch.nn.functional as F
import torch.optim as optim
from torchvision import datasets, transforms
from torch.optim.lr_scheduler import StepLR

class Net(nn.Module):
    def __init__(self):
        super(Net, self).__init__()
        self.conv1 = nn.Conv2d(1, 32, 3, 1)
        self.conv2 = nn.Conv2d(32, 64, 3, 1)
        self.dropout1 = nn.Dropout2d(0.25)
[...]
test
-----
import torch
import torch.nn as nn
import torch.nn.functional as F
from torchvision import datasets, transforms

# To run on GPU, replace 'cpu' by 'cuda'
device = torch.device('cpu')
MEm kill ring* L1 [15 Candidate(s)] C-h m:Help TAB:Act RET/f1/f2/f:n:NthAct C-l:Tog.suspend C-h c:C - .emacs 61% 2788 Emacs-Lisp s U: /home/marie/.emacs
pattern: |
```

# Navigate open & recent files

```
C-j: Show this buffer (keeping session)
Buffers
  _index.md
  sketch.org
  alias.zsh
  script.sh
  .emacs

Recentf
  .emacs
  move-westgrid-users.csv
  rifle.conf
  _index.md
  sketch.org
  .gitignore
  alias.zsh
  contact.org
  config
  .zshrc
  script.sh
  julia.jl
  bookmarks
  upscaling_data.org
  covid_vaccination_info.org
  config.py
  linux.org
  random.org
  passphrase.gpg
  .xinitrc
  fontlist-v330.json
  test.R
  mc.org
  julia-11-shared-arrays.md
  config
  monthly_script.sh
  weekly_script.sh

X Windows
st-256color. ranger:image/logo/lwg
Emacs .emacs
st-256color. marie@prosoitos:~
qutebrowser. 0 top Inbox - marie.burle@westgrid.ca - Compute Canada Inc Mail
st-256color. marie@prosoitos:~
st-256color. man grep
st-256color. man rg

Create buffer

(window-divider ((t (:foreground "gray25"))))
(yas-field-highlight-face ((t (:underline t)))))

;; * unset env variable MANPAGER in Emacs only
;; MANPAGER is set in zsh to sh -c 'col -bx | bat -l man -p'
;; which provides bat colour highlighting in the man pages
;; but this messes up the help files in inferior-python-mode

(setenv "MANPAGER" "")

;; * define new prefix keys
;; ** ;

(define-prefix-command 'ring-map)
(global-set-key (kbd ";") 'ring-map)

;; *** make sure I can still type ;
;; (global-set-key (kbd "; SPC")
;; (lambda () (interactive) (insert ";")))

;; (global-set-key (kbd "; SPC") 'ignore)

(global-set-key (kbd "; SPC") 'ignore)

(global-set-key (kbd ";" '")
(lambda () (interactive) (insert ";")))

;; ** C-z

(define-prefix-command 'ring-map-two)
(global-set-key (kbd "C-z") 'ring-map-two)

;; * setup

(require 'org) ; necessary

;; ** set default load paths

(let ((default-directory "~/emacs.d/elpa/"))
  (setq load-path
    (append
      (let ((load-path (copy-sequence load-path))) ;; Shadow
        (append
          (copy-sequence (normal-top-level-add-to-load-path '(".")))
          (normal-top-level-add-subdirs-to-load-path)))
        load-path)))

;; ** package system
```

#helm-taskswitch L1 [5 Buffer(s)] C-h m:Help TAB:Act RET/f1/f2/f-n:NthAct C-!:Toq.suspend C-h c:Con - .emacs 61% 2788 Emacs-Lisp s U: /home/marie/.emacs  
pattern: |

# Navigate recent files

```
C-j: Show this file (keeping session)
Recentf
~/emacs
~/parvus/ptmp/dl/move-westgrid-users.csv
~/config/ranger/rifle.conf
~/parvus/prog/slides/content/tools/_index.md
~/parvus/adm/notes/sketch.org
~/gitignore
~/zsh/custom/alias.zsh
~/parvus/adm/notes/contact.org
~/config/bat/config
~/zshrc
~/parvus/prog/slides/static/files/tools/script.sh
~/parvus/prog/slides/static/files/tools/julia.jl
~/emacs.d/bookmarks
~/parvus/prog/tml/content/webinars/upscaling_data.org
~/parvus/adm/medical/vaccination/covid/covid_vaccination_info.org
~/config/qutebrowser/config.py
~/parvus/adm/notes/linux.org
~/parvus/adm/notes/random.org
~/password-store/passphrase.gpg
~/xinitrc
~/cache/matplotlib/fontlist-v330.json
~/parvus/prog/progtest/test.R
~/parvus/adm/notes/mc.org
~/parvus/prog/courses/content/julia202202/julia-11-shared-arrays.md
~/config/i3/config
~/hbin/btimers/monthly_script.sh
~/hbin/btimers/weekly_script.sh

(global-set-key (kbd "; b s") 'bookmark-save)
;; **** jump to bookmark and recenter

(defun my-bookmark-jump (bookmark &optional display-func)
  (interactive)
  (list (bookmark-completing-read "Jump to bookmark"
                                  bookmark-current-bookmark)))

(unless bookmark
  (error "No bookmark specified"))
(bookmark-maybe-historicize-string bookmark)
(bookmark-jump-via bookmark (or display-func 'pop-to-buffer-same-window))
(recenter-top-bottom))

(global-set-key (kbd "; e k") 'my-bookmark-jump)

;; ** goto directory of current buffer

(fset 'my-go-to-current-directory
      [?\M-x ?i ?d ?o ? return return])

(global-set-key (kbd "; C-l") 'my-go-to-current-directory)

;; ** goto github

(global-set-key (kbd "; h v f") 'helm-open-github-from-file) ; v = version (for version
control)

(global-set-key (kbd "; h v c") 'helm-open-github-from-commit)

;; ** search regexp in content

;; ** within current buffer

(defun my-helm-occur ()
  (interactive)
  (helm--advice-push-mark)
  (helm-occur))

(global-set-key (kbd "; j") 'my-helm-occur)

;; ** within open buffers (ag)
;; search within content of buffers

;; **** all open buffers

(global-set-key (kbd "; h a b") 'helm-do-ag-buffers) ; helm buffers

;; **** open org buffers

(global-set-key (kbd "; h a o") 'helm-do-ag-org-buffers)

#helm recentf* L1 [27 File(s)] C-h m:Help TAB:Act RET/f1/f2/f-n:NthAct C-!:Tog.suspend C-h c:Conf
pattern: |
- .emacs 66% 3313 Emacs-Lisp s U: /home/marie/.emacs
```

# Find file

```
C-]: Hit1 Expand Candidate, Hit2 or (C-u) Find file (keeping session)
Read File Name (C-l: Go up one level)
/home/marie/.
/home/marie/..
.Rprofile
.Xauthority
.Xresources
.abcde.conf
.android
.aspell.en.prepl
.aspell.en.pws
.aspell.fr.prepl
.aspell.fr.pws
.authinfo.gpg
.bash_profile
.bashrc
.borg
.cache
.cargo
.cht.sh
.config
.emacs
.emacs-bmk-bmenu-state.el
.emacs.d
.fasd
.git
.gitconfig
.gitignore
.gnome
.gnupg
.googleearth
.gtkrc-2.0
.gtkrc-2.0.mine
.icons
.inputrc
.ipython
.julia
.jupyter
.less
.lessshst
.lesskey
.llpp_my_kbd.ml
.llpp_original_kbd.ml
.local
.mbsyncrc
.msmtpl.log
.msmtprc
.npm
.nv
.nvidia-settings-rc
.password-store
#helm file completions* L1 [92 Candidate(s)] C-h m:Help C/M-RET:Empty TAB:Act RET/f1/f2/f-n:NthAct C-l : .emacs 66% 3347 Emacs-Lisp s U: /home/marie/.emacs
Search in file(s): /home/marie/

(helm--advice-push-mark)
(helm-occur))

(global-set-key (kbd "; j") 'my-helm-occur)

;; *** within open buffers (ag)
;; search within content of buffers

;; **** all open buffers

(global-set-key (kbd "; h a b") 'helm-do-ag-buffers) ; helm buffers

;; **** open org buffers

(global-set-key (kbd "; h a o") 'helm-org-rifle)

;; *** within files (ag)
;; search within content of files (no need to be open)

;; **** in directory

(global-set-key (kbd "; h a d") 'helm-do-ag)
;; defaults to current directory, but path can be edited

;; **** org files in directory

(global-set-key (kbd "; h a r") 'helm-org-rifle-directories)
;; with prefix: recursively

;; **** in current project

(global-set-key (kbd "; h a p") 'helm-do-ag-project-root) ; helm project

;; ** search file names

;; *** current directory
;; search files in current directory
;; directory can be changed

(global-set-key (kbd "; f f") 'helm-find-files)

;; *** open buffers
;; search among buffer names (not content)

(global-set-key (kbd "; i") 'helm-taskswitch)

(require 'helm-taskswitch) ; not necessary to load the package, but necessary
for the function below not to break it

(defun helm-taskswitch ()
```

# Locate

```
C-j: helm-ff-kill-or-find-buffer-fname (Keeping session)
Locate
/boot/grub/x86_64-efi/configfile.mod
/etc/abcde.conf
/etc/conf.d
/etc/daxctl.conf.d
/etc/dconf
/etc/dhcpd.conf
/etc/e2scrub.conf
/etc/fuse.conf
/etc/gai.conf
/etc/grc.conf
/etc/healthd.conf
/etc/host.conf
/etc/i3status.conf
/etc/krb5.conf
/etc/ld.so.conf
/etc/ld.so.conf.d
/etc/libao.conf
/etc/libaudit.conf
/etc/libva.conf
/etc/locale.conf
/etc/makepkg.conf
/etc/man_db.conf
/etc/mdadm.conf
/etc/mke2fs.conf
/etc/mkinitcpio.conf
/etc/ndctl.conf.d
/etc/netconfig
/etc/nscd.conf
/etc/nsswitch.conf
/etc/ntp.conf
/etc/pacman.conf
/etc/request-key.conf
/etc/resolv.conf
/etc/resolv.conf.bak
/etc/resolvconf.conf
/etc/sensors3.conf
/etc/sudo.conf
/etc/sudo_logsrvd.conf
/etc/ts.conf
/etc/updatedb.conf
/etc/xattr.conf
/etc/R/Makeconf
/etc/R/javaconf
/etc/UPower/UPower.conf
/etc/X11/xorg.conf.d
/etc/audit/auditd-remote.conf
/etc/audit/auditd.conf
/etc/audit/zos-remote.conf
/etc/avahi/avahi-daemon.conf
helm locate* [1] [3496 File(s)] C-h m:Help TAB:Act RET/f1/f2/f-n:NthAct C-!:Toq.suspend C-h c:Conf
pattern: conf

;; *** current directory
;; search files in current directory
;; directory can be changed

(global-set-key (kbd "; f f") 'helm-find-files)

;; *** open buffers
;; search among buffer names (not content)

(global-set-key (kbd "; i") 'helm-taskswitch)

(require 'helm-taskswitch) ; not necessary to load the package, but necessary
for the function below not to break it

(defun helm-taskswitch ()
  "Use helm to switch between tasks (X11 windows, buffers or recentf)."
  (interactive)
  (run-hooks 'helm-taskswitch-open-hooks)
  (select-frame-set-input-focus (window-frame (selected-window)))
  (unless helm-source-buffers-list
    (setq helm-source-buffers-list
          (helm-make-source "Buffers" 'helm-source-buffers)))
  (helm :sources '(helm-source-buffers-list
                  helm-source-recentf
                  helm-taskswitch-source-x-windows
                  helm-source-buffer-not-found)
        :buffer "*helm-taskswitch*"
        :truncate-lines t))

;; *** current git project

(global-set-key (kbd "; h p") 'helm-cmd-t)

;; *** git of files in current git project
;; using helm-ls-git package

(global-set-key (kbd "; h g") 'helm-browse-project)

;; *** all files in computer
;; equivalent to: locate -i -e -A --regex
;; regex

(global-set-key (kbd "; h l") 'helm-locate)

;; ** search email

(global-set-key (kbd "; m s") 'helm-mu) ; mu search

(global-set-key (kbd "; m c") 'helm-mu-contacts)

;; ** search help

helm locate* [1] [3496 File(s)] C-h m:Help TAB:Act RET/f1/f2/f-n:NthAct C-!:Toq.suspend C-h c:Conf - .emacs 66% 3383 Emacs-Lisp s U: /home/marie/.emacs
```

# Search objects in working environment

```
C-j: Show this entry (keeping session)
Imenu
Types / my-shortcode-face
Types / my-nested-org-block-face
Variables / aw-dispatch-alist
Function / custom set variables
Function / custom set faces
Function / unset env variable MANPAGER in Emacs only
Function / define new prefix keys
Function / ;
Function / make sure I can still type ;
Function / C-z
Function / setup
Function / set default load paths
Function / package system
Function / prevent series of error messages in minibuffer due to slack
Function / change yes-no questions to y-n
Function / allow a selected area to be overwritten
Function / start emacs server
Function / emojiify emacs lisp mode
Function / font size
Function / frame
Function / create new frames without scrollbars and in fullscreen
Function / my/disable-scroll-bars
Function / disable menubar
Function / set frame title to buffer name
Function / kill buffer when frame is killed
Function / my-kill-buffer-with-frame
Function / explicitly load helm-mode early to prevent new error message
Function / fix word
Function / headings
Function / unicode
Function / dot
Function / m-dash
Function / n-dash
Function / degree
Function / Spanish punctuation
Function / single quotes
Function / double quotes
Function / French quotes
Function / French single quotes
Function / arrows
Function / mathematical signs
Function / beginning of text on line
Function / reload buffer after file modification without confirmation
Function / revert-buffer-no-confirm
Function / ido and smex
Function / ido completing read+ and ido ubiquitous
Function / ido completing for some functions
Function / ido vertical mode
Function / fix ido
helm-imenu L6 (HF) [100 Candidate(s)] C-h m:Help TAB:Act RET/f1/f2/f-n:NthAct C-l:Log.suspend C-h c:- .emacs 61% 2788 Emacs-Lisp s U: /home/marie/.emacs
pattern:
```

```
'(window-divider ((t (:foreground "gray25"))))
'(yas-field-highlight-face ((t (:underline t)))))

;; * unset env variable MANPAGER in Emacs only
;; MANPAGER is set in zsh to sh -c 'col -bx | bat -l man -p'
;; which provides bat colour highlighting in the man pages
;; but this messes up the help files in inferior-python-mode

(setenv "MANPAGER" "")

;; * define new prefix keys
;; ** ;

(define-prefix-command 'ring-map)
(global-set-key (kbd ";") 'ring-map)

;; *** make sure I can still type ;
;; (global-set-key (kbd "; SPC")
;;   (lambda () (interactive) (insert ";")))
;; (global-set-key (kbd "; SPC") 'ignore)

(global-set-key (kbd "; SPC") 'ignore)

(global-set-key (kbd ";" " ")
  (lambda () (interactive) (insert ";")))

;; ** C-z

(define-prefix-command 'ring-map-two)
(global-set-key (kbd "C-z") 'ring-map-two)

;; * setup
(require 'org) ; necessary

;; ** set default load paths

(let ((default-directory "~/emacs.d/elpa/"))
  (setq load-path
        (append
          (let ((load-path (copy-sequence load-path))) ;; Shadow
            (append
              (copy-sequence (normal-top-level-add-to-load-path '(".")))
              (normal-top-level-add-subdirs-to-load-path)))
          load-path)))

;; ** package system
```

# Search in Git repo

```
C-j: Diff (keeping session)
Git status (m)
M .aspell.en.pws
M .config/llpp.conf
M .config/qutebrowser/config.py
M .config/qutebrowser/quickmarks
M .config/ranger/rc.conf
M .config/ranger/rifle.conf
M .emacs
M .zsh/custom/alias.zsh
?? .config/bat/
?? .config/ranger/initial_rifle.conf

Git branches
* m: a4141ef ranger: important commit: massive changes to rifle.conf: the file setting op

Buffers in git project (m)
_index.md
sketch.org
alias.zsh
script.sh
.emacs

Git files (m)
.config/autokey/data/phrase/1.txt
.config/autokey/data/phrase/archive/smileys/1.txt
.config/autokey/data/phrase/archive/smileys/2.txt
.config/autokey/data/phrase/archive/smileys/3.txt
.config/autokey/data/phrase/archive/smileys/4.txt
.config/autokey/data/phrase/archive/smileys/5.txt
.config/autokey/data/phrase/archive/smileys/D.txt
.config/autokey/data/phrase/archive/smileys/P.txt
.config/autokey/data/phrase/archive/smileys/o.txt
.config/autokey/data/phrase/archive/animal/pp.txt
.config/autokey/data/phrase/archive/animal/sc.txt
.config/autokey/data/phrase/archive/animal/st.txt
.config/autokey/data/phrase/archive/animal/ts.txt
.config/autokey/data/phrase/craigslist/cl.txt
.config/autokey/data/phrase/date_and_time/day.py
.config/autokey/data/phrase/expression/eg.txt
.config/autokey/data/phrase/expression/er.txt
.config/autokey/data/phrase/acronym/hpc.txt
.config/autokey/data/phrase/archive/animal/vta.txt
.config/autokey/data/phrase/archive/animal/vtu.txt
.config/autokey/data/phrase/archive/names/dov.txt
.config/autokey/data/phrase/archive/names/mag.txt
.config/autokey/data/phrase/email_address/sfu.txt
.config/autokey/data/phrase/emoji_for_emacs_slack_reactions/boo.txt
.config/autokey/data/phrase/expression/bcp.txt
.config/autokey/data/phrase/expression/btw.txt
.config/autokey/data/phrase/acronym/DHSL.txt
helm ls git+ L1 [10 Candidate(s)] C-h m:Help TAB:Act RET/f1/f2/f-n:NhAct C-l:To: suspend C-h c:Conf
pattern:

;; *** current directory
;; search files in current directory
;; directory can be changed

(global-set-key (kbd "; f f") 'helm-find-files)

;; *** open buffers
;; search among buffer names (not content)

(global-set-key (kbd "; i") 'helm-taskswitch)

(require 'helm-taskswitch) ; not necessary to load the package, but necessary
for the function below not to break it

(defun helm-taskswitch ()
  "Use helm to switch between tasks (X11 windows, buffers or recentf)."
  (interactive)
  (run-hooks 'helm-taskswitch-open-hooks)
  (select-frame-set-input-focus (window-frame (selected-window)))
  (unless helm-source-buffers-list
    (setq helm-source-buffers-list
          (helm-make-source "Buffers" 'helm-source-buffers)))
  (helm :sources '(helm-source-buffers-list
                  helm-source-recentf
                  helm-taskswitch-source-x-windows
                  helm-source-buffer-not-found)
        :buffer "*helm-taskswitch*"
        :truncate-lines t))

;; *** current git project

(global-set-key (kbd "; h p") 'helm-cmd-t)

;; *** git of files in current git project
;; using helm-ls-git package

(global-set-key (kbd "; h g") 'helm-browse-project)

;; *** all files in computer
;; equivalent to: locate -i -e -A --regex
;; regex

(global-set-key (kbd "; h l") 'helm-locate)

;; ** search email

(global-set-key (kbd "; m s") 'helm-mu) ; mu search

(global-set-key (kbd "; m c") 'helm-mu-contacts)

;; ** search help

helm 66% 3383 Emacs-Lisp s U: /home/marie/.emacs
```



# Manage major/minor modes

```
C-j: Describe major mode (keeping session)
Major mode
emacs-lisp-mode

Active minor modes
abbrev-mode
adaptive-wrap-prefix-mode
aggressive-indent-mode
async-bytecomp-package-mode
auto-composition-mode
auto-compression-mode
auto-encryption-mode
blink-cursor-mode
company-flx-mode
company-mode
company-quickhelp-local-mode
company-quickhelp-mode
crm-custom-mode
delete-selection-mode
eldoc-mode
electric-indent-mode
emojify-mode
emojify-mode-line-mode
file-name-shadow-mode
flx-ido-mode
flyspell-mode
font-lock-mode
global-aggressive-indent-mode
global-company-mode
global-eldoc-mode
global-emojify-mode
global-emojify-mode-line-mode
global-font-lock-mode
global-undo-tree-mode
ido-everywhere
ido-ubiquitous-mode
ido-vertical-mode
mouse-wheel-mode
outline-minor-mode
outshine-mode
pyvenv-mode
rainbow-delimiters-mode
rainbow-identifiers-mode
recentf-mode
shell-dirtrack-mode
show-paren-mode
smartparens-global-mode
smartparens-mode
telega-active-locations-mode
telega-patrons-mode
telega-root-auto-fill-mode
#Helm Describe Modes L1 [1 Candidate(s)] C-h m:Help TAB:Act RET/f1/f2/f-n:NthAct C-!:Tog.suspend C-!-.emacs 67% 3422 Emacs-Lisp s U: /home/marie/.emacs
pattern:
```

# Search help files

```
C-]: Toggle describe command (keeping session)
Commands
helm-ff-RET
helm-ff-TAB
helm-ag-mode
helm-epa-mode
helm-mark-all
helm-locate C-x c l
helm-info-bfd
helm-info-cpp
helm-info-ksba
helm-info-gzip
helm-info-wget
helm-info-libc
helm-info-gperf
helm-info-gpgme
helm-info-gpgrt
helm-info-sieve
helm-info-ediff
helm-world-time
helm-info-nettle
helm-info-eshell
helm-info-gccint
helm-info-R-lang
helm-follow-mode
helm-sexp-eval-1
helm-info-dired-x
helm-info-gettext
helm-info-libidn2
helm-ag-mode-jump
helm-ag-this-file
helm-swap-windows
helm-ff-icon-mode
helm-info-gsl-ref
helm-narrow-window
helm-info-readline
helm-info-autoconf
helm-keyboard-quit
helm-info-libtasn1
helm-adaptive-mode
helm-manage-advice
helm-select-xfont C-x c F
helm-grep-mode-jump
helm-info-transient
helm-info-inetutils
helm-cmd-t-run-grep
helm-info-nxml-mode
helm-enlarge-window
helm-ff-run-grep-ag
helm-ff-run-pdfgrep
helm-buffers-list C-x c C-x C-b
#helm apropos* L1 [100 Candidate(s)] C-h m:HelP TAB:Act RET/f1/f2/f-n:NthAct C-!:Tog.suspend C-h c:Con - .emacs 67% 3422 Emacs-Lisp s U: /home/marie/.emacs
pattern: helm
(global-set-key (kbd ";" h l) 'helm-locate)
;; ** search email
(global-set-key (kbd ";" m s) 'helm-mu) ; mu search
(global-set-key (kbd ";" m c) 'helm-mu-contacts)
;; ** search help
;; *** goto the code of the function at point
;; already defined M-.
;; go back with M-,
;; ** info
(global-set-key (kbd ";" h h i) 'helm-info) ; helm help info
;; ** function description
(global-set-key (kbd ";" h h a) 'helm-apropos) ; helm help apropos
;; ** mode
;; **** describe mode and helm describe modes
(global-set-key (kbd ";" <f1> m) 'describe-mode)
(define-key dired-mode-map (kbd ";" <f1> m) 'describe-mode)
(global-set-key (kbd "<f1> m") 'helm-describe-modes)
(define-key dired-mode-map (kbd "<f1> m") 'helm-describe-modes)
;; **** helm mode manager
(global-set-key (kbd ";" h m s) 'helm-switch-major-mode)
(global-set-key (kbd ";" h m e) 'helm-enable-minor-mode)
(global-set-key (kbd ";" h m d) 'helm-disable-minor-mode)
;; ** keybinding
(global-set-key (kbd ";" h b") 'helm-descbinds)
;; ** package
;; necessary to load it before the function below is loaded (otherwise it is loaded only
;; when used and the function below is replaced by the default function)
```

# Search key bindings

```
C-j: helm-descbinds-action:describe (keeping session)
Major Mode Bindings:
C-x          Prefix Command
ESC          Prefix Command
DEL          backward-delete-char-untabify
C-x C-a      Prefix Command
C-M-q       indent-pp-sexp
C-M-x       eval-defun
C-x C-a C-c  edebug-go-mode
C-x C-a C-l  edebug-where
C-x C-a RET  edebug-set-initial-mode
C-x C-a C-n  edebug-next-mode
C-x C-a C-s  edebug-step-mode

'tracking-mode' Minor Mode Bindings:
C-c          Prefix Command
C-c C-@      tracking-next-buffer
C-c C-SPC    tracking-next-buffer

'flyspell-mode' Minor Mode Bindings:
C-c          Prefix Command
C-,         flyspell-goto-next-error
C-.         flyspell-auto-correct-word
<ignore>    flyspell-auto-correct-previous-word
C-c $       flyspell-correct-word-before-point

'outshine-mode' Minor Mode Bindings:
TAB          outshine-kbd-TAB
ESC          Prefix Command
<M-S-down>  outshine-kbd-M-S-<down>
<M-S-left>  outshine-kbd-M-S-<left>
<M-S-right> outshine-kbd-M-S-<right>
<M-S-up>    outshine-kbd-M-S-<up>
<M-down>    outline-next-visible-heading
<M-left>    outshine-kbd-M-<left>
<M-right>   outshine-kbd-M-<right>
<M-up>     outline-previous-visible-heading
<backtab>   outshine-kbd-<backtab>
<remap>     Prefix Command
C-M-i       outshine-cycle-buffer
M-RET       outshine-kbd-M-RET
<remap> <self-insert-command> outshine-self-insert-command

'outline-minor-mode' Minor Mode Bindings:
C-c          Prefix Command
ESC          Prefix Command
;           Prefix Command
C-;         org-cycle
M-n         outline-next-visible-heading
M-p         outline-previous-visible-heading
; |         outshine-cycle-buffer
#helm-descbinds* L1 [1] candidate(s) C-h m:Help TAB:Act RET/f1/f2/f-n:NthAct C-!:Tog.suspend C-h c:Conf
pattern: |
```

# Search in emails

```
C-j: helm-mu-persistent-action (keeping session)
Search email with mu
22-02-21 Mountain Conditions Re... mhburle@gmail.com [MCR] Asulkan Valley - Sapphire Col to Dome
22-02-06 Mountain Conditions Re... mhburle@gmail.com [MCR] Avalanche Conditions Summary - Central Selkirk Ranges - Observation from this
22-02-04 Mountain Conditions Re... mhburle@gmail.com [MCR] Slocan Ranges - Snowpack and ski conditions
22-01-28 Mountain Conditions Re... mhburle@gmail.com [MCR] Avalanche Crest - Rogers Pass - Avalanche Crest #3 - Something to keep in mi
21-05-06 Mountain Conditions Re... mhburle@gmail.com [MCR Regional Summary] ACMG Mountain Conditions Report for the Rockies and Columbia
20-11-12 Mountain Conditions Re... mhburle@gmail.com [MCR] Balu Pass, Glacier NP - Crusts galore at Rogers Pass
20-11-12 Mountain Conditions Re... mhburle@gmail.com [MCR] Balu Pass, Glacier NP - Crusts galore at Rogers Pass

#helm mu* [Mu Process Finish: (7 results)]
pattern: f:ountain conditions report
```

# Select colour

```
C-j: Kill entry in RGB format. (keeping session)
Colors
black                black                #000000 (global-set-key (kbd "; h m s") 'helm-switch-major-mode)
dim gray             dim gray,dim grey,DimGray,DimGrey      #696969 (global-set-key (kbd "; h m e") 'helm-enable-minor-mode)
dark gray           dark gray,dark grey,DarkGray,DarkGrey #a9a9a9 (global-set-key (kbd "; h m d") 'helm-disable-minor-mode)
gray                gray,gray                #bebebe
light gray          light gray,light grey,LightGray,LightGrey #d3d3d3
gainsboro           gainsboro                 #dcdcdc
white smoke         white smoke,WhiteSmoke  #f5f5f5
white               white                    #ffffff
red                 red                      #ff0000
orange red          orange red,OrangeRed     #ff4500
dark orange         dark orange,DarkOrange  #ff8c00
orange              orange                   #ffa500
gold                gold                     #ffd700
yellow              yellow                    #ffff00
chartreuse          chartreuse                #7fff00
lawn green          lawn green,LawnGreen     #7cfc00
green               green                     #00ff00
spring green        spring green,SpringGreen #00ff7f
medium spring green medium spring green,MediumSpringGreen #00ffa9a
cyan                cyan                      #00ffff
deep sky blue       deep sky blue,DeepSkyBlue #00bfff
blue                blue                      #0000ff
medium blue         medium blue,MediumBlue   #0000cd
dark violet         dark violet,DarkViolet   #4800d3
dark magenta        dark magenta,DarkMagenta #8b008b
magenta             magenta                   #ff00ff
dark red            dark red,DarkRed         #8b0000
brown               brown                     #a52a2a
firebrick           firebrick                 #b22222
indian red          indian red,IndianRed     #cd5c5c
light coral         light coral,LightCoral   #f08080
salmon              salmon                    #fa8072
light salmon        light salmon,LightSalmon #ffa07a
tomato              tomato                    #ff6347
coral               coral                      #ff7f50
dark salmon         dark salmon,DarkSalmon   #e9967a
rosy brown          rosy brown,RosyBrown     #bc8f8f
sienna              sienna                    #a0522d
saddle brown        saddle brown,SaddleBrown #8b4513
chocolate           chocolate                 #d2691e
peru                peru                      #cd853f
sandy brown         sandy brown,SandyBrown   #f4a460
burlywood           burlywood                 #deb887
tan                 tan                       #d2b48c
navajo white        navajo white,NavajoWhite #ffdead
wheat               wheat                     #f5deb3
dark goldenrod      dark goldenrod,DarkGoldenrod #b8860b
goldenrod           goldenrod                 #daa520
light goldenrod     light goldenrod,LightGoldenrod #eedd82
; ** keybinding
(global-set-key (kbd "; h b") 'helm-descbinds)
; ** package
; necessary to load it before the function below is loaded (otherwise it is loaded only
; when used and the function below is replaced by the default function)
(require 'helm-system-packages)
(defun helm-system-packages-pacman-uninstall (_)
  "Uninstall marked candidates."
  (helm-system-packages-run-as-root-over-installed
   "pacman" "--Rsn"
   (when helm-current-prefix-arg "--recursive")
   (unless helm-system-packages-pacman-confirm-p "--noconfirm")))
(global-set-key (kbd "; h h p") 'helm-system-packages)
; ** color
(global-set-key (kbd "; h c") 'helm-colors)
; ** search web engines with surfraw
(global-set-key (kbd "; h w") 'helm-surfraw) ; helm web
; * helm M-x
(global-set-key (kbd "; M-x") 'helm-M-x)
; * undo/redo
(global-set-key (kbd "C-l") 'undo)
; ** undo tree
(global-set-key (kbd "s-t") 'undo-tree-visualize)
; go to selected undo state
(define-key undo-tree-visualizer-mode-map (kbd "<return>") 'undo-tree-visualizer-quit)
; cancel (return to state before calling undo-tree-visualize)
(define-key undo-tree-visualizer-mode-map (kbd "q") 'undo-tree-visualizer-abort)
helm colors* L1 100 Candidate(s) C-h m:Help TAB:Act RET/f1/f2/f-n:NthAct C-l:Toggle.suspend C-h c:Conf - .emacs 67% 3456 Emacs-Lisp s U: /home/marie/.emacs
pattern: |
```

# Select emoji

```
C-j: Insert Character (keeping session)
unicode-characters
FACE PALM 🤦
FACE MASSAGE 🤦
FACE WITH MONOCLE 🧐
FACE WITHOUT MOUTH 😬
FACE THROWING A KISS 😘
FACE WITH COLD SWEAT 🥵
FACE WITH COWBOY HAT 🤠
FACE WITH OK GESTURE 🙆
FACE WITH OPEN MOUTH 🤔
FACE WITH THERMOMETER 🌡
FACE SCREAMING IN FEAR 😱
FACE WITH HEAD-BANDAGE 🩹
FACE WITH MEDICAL MASK 🧑
FACE WITH ROLLING EYES 🤔
FACE WITH TEARS OF JOY 😂
FACE WITH PLEADING EYES 🙏
FACE WITH LOOK OF TRIUMPH 😏
FACE WITH NO GOOD GESTURE 🙄
FACE WITH STUCK-OUT TONGUE 🙄
FACE WITH ONE EYEBROW RAISED 🙄
FACE SAVOURING DELICIOUS FOOD 🍴
FACE WITH OPEN MOUTH VOMITING 🤮
FACE WITH PARTY HORN AND PARTY HAT 🎉
FACE WITH OPEN MOUTH AND COLD SWEAT 🥵
FACE WITH UNEVEN EYES AND WAVY MOUTH 🤪
FACE WITH FINGER COVERING CLOSED LIPS 🤔
FACE WITH STUCK-OUT TONGUE AND WINKING EYE 🙄
FACE WITH STUCK-OUT TONGUE AND TIGHTLY-CLOSED EYES 🙄
CAT FACE 🐱
COW FACE 🐮
DOG FACE 🐶
FOX FACE 🦊
BEAR FACE 🐻
FROG FACE 🐸
ANGRY FACE 😡
CLOWN FACE 🤡
DIZZY FACE 🤯
HORSE FACE 🐎
CRYING FACE 😭
DRAGON FACE 🐉
HUSHED FACE 🤫
FEARFUL FACE 😨
FLUSHED FACE 😳
GIRAFFE FACE 🦒
HAMSTER FACE 🐹
HUGGING FACE 🤗
CONFUSED FACE 😕
#helm-unicode-search L1 [100 Candidate(s)] C-h m:HelP TAB:Act RET/f1/f2/f-n:NthAct C-!:Tog.suspend C-! .emacs 61% 2788 Emacs-Lisp s U: /home/marie/.emacs
pattern: face
```

**Questions?**