

Emacs Prelude Cheat Sheet

Getting Help in Emacs

<code>C-h k</code>	<code><key></code>	describe function bound to <code><key></code>
<code>C-h b</code>		list key-bindings for focused buffer
<code>C-h m</code>		describe mode
<code>C-h l</code>		show the keys you have pressed
<code>C-h f</code>		describe function

Global

<code>C-x \</code>		align-regexp
<code>C++</code>		text-scale-increase
<code>C--</code>		text-scale-decrease
<code>C-x 0</code>		go to previous window
<code>C-x m</code>		eshell
<code>C-x M</code>		start a new eshell if one is active
<code>C-x M-m</code>		shell
<code>C-x C-m</code>		smex
<code>C-h A</code>		apropos
<code>C-h C-f</code>		find-function
<code>C-h C-k</code>		find-function-on-key
<code>C-h C-v</code>		find-variable
<code>C-h C-l</code>		find-library
<code>C-h C-i</code>		info-display-manual
<code>C-<backspace></code>		kill line backward and indent
<code>C-o</code>		(isearch-mode) isearch-occur
<code>M-/</code>		hippie-expand
<code>C-x C-b</code>		ibuffer
<code><f12></code>		menu-bar-mode
<code>C-x g</code>		magit-status
<code>C-x M-g</code>		magit-dispatch
<code>C-c g</code>		magit-file-dispatch

prelude

Keybindings defined in prelude mode

<code>C-c o</code>		crux-open-with
<code>C-c C-/ g</code>		prelude-google
<code>C-c C-/ h</code>		prelude-github
<code>C-c C-/ y</code>		prelude-youtube
<code>C-c C-/ d</code>		prelude-duckduckgo
<code>C-a</code>		crux-move-beginning-of-line
<code>S-<return></code>		crux-smart-open-line
<code>M-o</code>		crux-smart-open-line
<code>C-S-<return></code>		crux-smart-open-line-above
<code>C-S-<up></code>		move-text-up
<code>C-S-<down></code>		move-text-down
<code>M-S-<up></code>		move-text-up
<code>M-S-<down></code>		move-text-down
<code>C-c n</code>		crux-cleanup-buffer-or-region
<code>C-c f</code>		crux-recentf-ido-find-file
<code>C-M-z</code>		crux-indent-defun
<code>C-c u</code>		crux-view-url
<code>C-c e</code>		crux-eval-and-replace
<code>C-c s</code>		crux-swap-windows
<code>C-c w</code>		prelude-swap-meta-and-super

<code>C-c D</code>		crux-delete-file-and-buffer
<code>C-c d</code>		crux-duplicate-current-line-or-region
<code>C-c M-d</code>		crux-duplicate-and-comment-current- line-or-region
<code>C-c r</code>		crux-rename-buffer-and-file
<code>C-c t</code>		crux-visit-term-buffer
<code>C-c k</code>		crux-kill-other-buffers
<code>C-c TAB</code>		crux-indent-rigidly-and-copy- to-clipboard
<code>C-c I</code>		crux-find-user-init-file
<code>C-c S</code>		crux-find-shell-init-file
<code>C-c i</code>		imenu-anywhere
<code>s-p</code>		projectile-command-map
<code>s-r</code>		crux-recentf-ido-find-file
<code>s-j</code>		crux-top-join-line
<code>s-k</code>		crux-kill-whole-line
<code>s-m m</code>		magit-status
<code>s-m l</code>		magit-log
<code>s-m f</code>		magit-log-buffer-file
<code>s-m b</code>		magit-blame
<code>s-o</code>		crux-smart-open-line-above

ace-window

Quickly switch windows in Emacs

<code>s-w</code>		ace-window
avy		
Effectively navigate to visible things		
<code>C-c v</code>		avy-goto-word-or-subword-1
<code>s-.</code>		avy-goto-word-or-subword-1

anzu

Enhances isearch & query-replace by showing total matches and current match position		
<code>M-%</code>		anzu-query-replace
<code>s-.</code>		anzu-query-replace-regexp

browse-kill-ring

Smarter kill-ring navigation

<code>s-y</code>		browse-kill-ring
crux		
A Collection of Ridiculously Useful eXtensions for Emacs.		
<code>C-^</code>		crux-top-join-line
<code>s-k</code>		crux-kill-whole-line

discover-my-major

A quick major mode help with discover-my-major.

<code>C-h C-m</code>		discover-my-major
easy-kill		
Kill & Mark Things Easily in Emacs.		
<code>M-w</code>		easy-kill
<code>C-M-SPC</code>		easy-mark
<code>C-M-@</code>		easy-mark

expand-region

Increase selected region by semantic units.

<code>C-=</code>		er/expand-region
------------------	--	------------------

flycheck

On the fly syntax checking.

<code>C-t C-n</code>		flycheck-next-error
<code>C-t C-p</code>		flycheck-previous-error
projectile		
Project Interaction Library for Emacs.		
<code>C-c p f</code>		projectile-find-file
<code>C-c p p</code>		projectile-switch-project
<code>C-c p d</code>		projectile-find-dir
<code>C-c p r</code>		projectile-replace
<code>C-c p T</code>		projectile-find-test-file
<code>C-c p s g</code>		projectile-grep
<code>C-c p s s</code>		projectile-ag

operate-on-number

Operate on number at point with arithmetic functions.

<code>C-c . +</code>		apply-operation-to-number-at-point
<code>C-c . -</code>		apply-operation-to-number-at-point
<code>C-c . *</code>		apply-operation-to-number-at-point
<code>C-c . /</code>		apply-operation-to-number-at-point
<code>C-c . \</code>		apply-operation-to-number-at-point
<code>C-c . ^</code>		apply-operation-to-number-at-point
<code>C-c . <</code>		apply-operation-to-number-at-point
<code>C-c . ></code>		apply-operation-to-number-at-point
<code>C-c . #</code>		apply-operation-to-number-at-point
<code>C-c . %</code>		apply-operation-to-number-at-point
<code>C-c . '</code>		operate-on-number-at-point

zop-to-char

A more powerful alternative to zap-to-char.

<code>M-z</code>		zop-up-to-char
<code>M-Z</code>		zop-to-char

Modules

prelude-ido

<code>M-x</code>		smex
<code>M-X</code>		smex-major-mode-commands

prelude-ivy

<code>C-c C-r</code>		ivy-resume
<code><f6></code>		ivy-resume
<code>C-s</code>		swiper
<code>M-x</code>		counsel-M-x
<code><f1> f</code>		counsel-describe-function
<code><f1> v</code>		counsel-describe-variable
<code><f1> l</code>		counsel-find-library
<code><f1> i</code>		counsel-info-lookup-symbol
<code><f1> u</code>		counsel-unicode-char
<code>C-c g</code>		counsel-git
<code>C-c j</code>		counsel-git-grep
<code>C-c k</code>		counsel-ag
<code>C-x l</code>		counsel-locate
<code>C-r</code>		(minibuffer) counsel-minibuffer-history

prelude-helm

<code>C-c h o</code>		helm-occur
<code>C-c h g</code>		helm-do-grep
<code>C-c h C-c w</code>		helm-wikipedia-suggest
<code>C-c h SPC</code>		helm-all-mark-rings

<code>C-c p h</code>		helm-projectile
prelude-helm-everywhere		
<code>M-x</code>		helm-M-x
<code>C-c C-m</code>		helm-M-x
<code>M-y</code>		helm-show-kill-ring
<code>C-x b</code>		helm-mini
<code>C-x C-b</code>		helm-buffers-list
<code>C-x C-f</code>		helm-find-files
<code>C-h f</code>		helm-apropos
<code>C-h r</code>		helm-info-emacs
<code>C-h C-l</code>		helm-locate-library
<code>C-c f</code>		(prelude-mode) helm-recentf
<code>C-c C-l</code>		(minibuffer-local) help-minibuffer-history
<code>C-o</code>		(isearch-mode) helm-occur-from-isearch
<code>C-c C-l</code>		(shell-mode) helm-comint-input-ring

prelude-key-chord

<code>jj</code>		avy-goto-word-1
<code>j1</code>		avy-goto-line
<code>jk</code>		avy-goto-char
<code>JJ</code>		crux-switch-to-previous-buffer
<code>uu</code>		undo-tree-visualize
<code>xx</code>		execute-extended-command
<code>yy</code>		browse-kill-ring

prelude-evil

<code>C-A</code>		(normal-state) evil-numbers/inc-at-pt
<code>C-S-A</code>		(normal-state) evil-numbers/dec-at-pt
<code>></code>		(visual-state) prelude-shift-right-visual
<code><</code>		prelude-shift-left-visual
<code>C-S-d</code>		prelude-evil-scroll-down-other-window
<code>C-S-u</code>		prelude-evil-scroll-up-other-window
<code>K</code>		(magit-branch-manager-mode) (emacs-state) magit-discard
<code>L</code>		(magit-branch-manager-mode) (emacs-state) magit-log-popup
<code>K</code>		(magit-status-mode) (emacs-state) magit-discard
<code>l</code>		(magit-status-mode) (emacs-state) magit-log-popup
<code>h</code>		(magit-status-mode) (emacs-state) magit-diff-toggle-refine-hunk

prelude-emacs-lisp

<code>C-c C-z</code>		prelude-visit-ielm
<code>C-c C-c</code>		eval-defun
<code>C-c C-b</code>		eval-buffer

prelude-go

<code>C-c a</code>		go-test-current-project
<code>C-c m</code>		go-test-current-file
<code>C-c .</code>		go-test-current-test
<code>C-c b</code>		go-run
<code>C-c h</code>		godoc-at-point

prelude-org

<code>C-c l</code>		org-store-link
<code>C-c a</code>		org-agenda
<code>C-c b</code>		org-iswitchb