



Emacs for Clojure Cheatsheet

The barest minimum to get going with emacs for clojure development: More details at <http://bit.ly/flexiana-emacs>

Flexiana

TERMINOLOGY

M- the meta key. Usually defaults to alt/option
C- control key
S- shift key
(...) Anything in parentheses is lisp code, run it with M-S-;
nnn-*nnn* Kebab-case-names are emacs commands. Run them with C-x

INSTALL CIDER

Cider is not part of the default emacs install so add that by running these lisp commands

```
(require ,package)
(add-to-list ,package-archives
  ,(„melpa“ . „http://melpa.org/packages/“) t)
(package-install ,cider)
```

M-x cider-jack-in Start a repl in the current directory.

WINDOW CONTROLS.

C-x o (the letter o) Switch to other window
C-x 1 close all other windows
C-x 0 (digit 0) close this window
C-x 2 split window horizontally
C-x 3 split window vertically

OPENING FILES

C-x C-d show directory listing
<return> open file at point

NAVIGATION

C-e end of line
C-a start of line
C-n next line
C-p previous line
C-f forward
C-b back
C-v scroll screen down
M-v scroll screen up
 (arrow keys also work as expected. Ctrl-arrow gives bigger movements)

CUT AND PASTE

C-x space sets “mark”
C-w cut region to cut buffer
M-w copy region to buffer
C-y yank from buffer (everyone else calls this paste)
M-y cycle yanked region (ie paste earlier cuts)

HELP

C-h f *function-name* Describe what function name does.
C-h v *variable-name* Describe what a variable contains and what it controls.
C-h k *key-sequence* Describe the function bound to key sequence.
C-h b List the bindings for the current buffer.
C-k i Emacs info. n,p,u,d to navigate here.
C-k t Run the emacs tutorial.

OTHER ESSENTIALS

C-x C-s save
C-x C-c quit emacs
C-s search in file
C-g cancel
C-/ Undo
TAB show or make completions
M-S-; execute lisp function.
M-x <cmd> run cmd ie:
(not bound) M-x rgrep - search in many files

CIDER

cider-jack-in Starts nrepl in the current directory and connects to it.
cider-jack-in-cljs Same but starts a clojurescript repl.
cider-connect Connects to a running repl instance.
cider-eval-defn-at-point Evaluate the clojure form around the point.
cider-eval-last-sexp Evaluate the clojure form preceding the point.
cider-repl-set-ns Switch the repl to the namespace of the current file.

Flexiana

Team of senior developers and software testers
to your service **now**

Contact us: team@flexiana.com



AGILE

Openness,
transparency,
cooperation,
frequent
releases



SENIOR-ONLY

Experienced
people,
well organized,
disciplined,
effective



FULL REMOTE

We search for talent
globally currently
covering 14 countries,
predominantly
within the EU



CODEVELOPMENT

We manage
people & we
integrate to your
development
at the same time



HIGH QUALITY

We are people
who prefer quality
and maintainability

www.flexiana.com