

Doom configuration

David Lewis

September 30, 2022

Contents

1	Identity information	1
2	Doom config	1
3	Build information	1
4	Theme	1
5	Org Setup	1
5.1	Org Directory	1
5.2	Hide emphasis markers	2
5.3	Org Tab behaviour	2
5.4	org code blocks	2
5.5	Functions	2
5.6	Org default image size	2
5.7	org keymap	2
5.8	org headline	2
5.9	org-fancy-priorities-mode work around	3
5.10	org-safe-variables	3
	5.10.1 html export	3
	5.10.2 Latex Export	3
	5.10.3 Org-babel	3
5.11	format all error	3
5.12	gnuplot	3
5.13	org-fragtog	3
5.14	ispell	3
5.15	Startup	3
5.16	citations	4

5.17	set header args	4
5.18	org-xournalpp	5
5.19	Disable Line wrapping	5
5.20	Org pomodoro	5
5.21	nix hack	5
5.22	plantuml	5
6	Latex setup	5
6.1	Use LuaTex	5
6.2	Keymap	6
7	Spell setup	6
7.1	Personal Dictionary	6
7.2	Fix hunspell bug	6
8	Python setup	6
8.1	Anaconda directory	6
8.2	LSP nix	6
9	R setup	6
9.1	keymap	6
9.2	Font lock keywords	7
9.2.1	R-major-mode	7
9.2.2	R-inferior-mode	7
9.3	Custom Functions	7
10	General configuration	7
10.1	remove line numbers	7
10.2	Fix treemacs ace-window bug	7
10.3	deletes compilation buffer if successful (ignores python buffers)	7
10.4	ligatures	8
10.5	ispell dictionary	8
11	Key Map	8
11.1	General Buffers	8
11.2	Python mode map	8
11.3	prolog map	8
11.4	Doc-view mode map	8
11.5	mips mode map	8
11.6	haskell map	8

12 remote	9
13 Helpful info	9
13.1 Font variables	9
13.2 Useful customization functions	9

1 Identity information

```
[fontsize=]elisp (setq user-full-name "David Lewis" user-mail-address "davidalewis00@gmail.com")
```

2 Doom config

```
[fontsize=]elisp (setq! doom-private-dir " /dotfiles/.doom.d/")
```

3 Build information

```
[fontsize=]sh ./configure -with-json -with-imagemagick -with-xpm=ifavailable -with-native-compilation
```

4 Theme

```
set theme with doom-theme or load theme with load-theme [fontsize=]elisp
(setq doom-theme 'doom-base16) (setq doom-themes-treemacs-theme "doom-colors") (add-hook
'pdf-tools-enabled-hook 'pdf-view-themed-minor-mode)
```

5 Org Setup

5.1 Org Directory

```
[fontsize=]elisp (setq org-directory " /Dropbox/org/")
```

5.2 Hide emphasis markers

```
[fontsize=]elisp (setq org-hide-emphasis-markers 't)
```

5.3 Org Tab behaviour

```
[fontsize=]elisp (after! evil-org (setq org-tab-first-hook nil))
```

5.4 org code blocks

```
[fontSize=]elisp (defun insert-jupyter-python-block () "Inserts a python code block" (interactive) (progn (insert "+beginsrcjupyter - python : sessionpy" + endsrc"))(org - edit - special))
  (defun insert-jupyter-ess-block() "Inserts an ESS(R) code block" (interactive) (insert "+beginsrcjupyter - R : sessionR" + endsrc"))(org - edit - special))
  (defun insert-elisp-block() "inserts an elisp code block" (interactive) (insert "+beginsrccelisp" + endsrc"))(org - edit - special))
  (defun insert-generic-block() "inserts an elisp code block" (interactive) (insert "+beginsrc"(read-string "Entersrctype : ")"" + endsrc"))(org - edit - special))(defun insert-gnuplot-block()(interactive)(insert "+beginsrcgnuplot : resultsoutput : file"(read-string "Enterfilename : ")"" + endsrc"))(org - edit - special))

  (defun copy-down-x (x) "copies the current cell down by the universal arg" (interactive "P")
    (cl-loop repeat (or x 1) do (org-table-copy-down 0)) ) (defun universal-test (x) (interactive "P")
    (cl-loop repeat x do (message "hello"))) (message "done"))
```

5.5 Functions

```
[fontSize=]elisp (defun cache-address () (interactive) (progn (setq size (read-string "Enter the MM size: ")) (setq strunit (substring size -2 nil)) (if (eq (compare-strings strunit nil nil "GB" nil nil) 't) (message "true"))) ) )
```

5.6 Org default image size

```
[fontSize=]elisp (after! evil-org (setq org-image-actual-width 400))
```

5.7 org keymap

```
[fontSize=]elisp (map! (:after org :map org-mode-map :nvi "S-<return>" 'copy-down-x :leader (:prefix ("j" . "jupyter-source") :desc "jupyter-python" "p" 'insert-jupyter-python-block :desc "jupyter-R" "r" 'insert-jupyter-ess-block :desc "elisp" "e" 'insert-elisp-block :desc "gnuplot" "g" 'insert-gnuplot-block :desc "generic" "b" 'insert-generic-block)))
```

5.8 org headline

```
[fontSize=]elisp (setq org-fontify-todo-headline t)
```

5.9 org-fancy-priorities-mode work around

```
[fontSize=]elisp (defadvice! +org-dont-fontify-my-thangs-a (orig-fn rest args) :around '(org-superstar-mode org-fancy-priorities-mode) (let! (('font-lock-ensure 'ignore) ('font-lock-flush 'ignore) ('font-lock-fontify-buffer 'ignore)) (apply orig-fn args)))
```

5.10 org-safe-variables

5.10.1 html export

```
[fontsize=]elisp (add-to-list 'safe-local-eval-forms '(add-hook 'after-save-hook 'org-html-export-to-html t t)) (add-to-list 'safe-local-eval-forms '(add-hook 'after-save-hook 'org-re-reveal-export-to-html t t))
```

5.10.2 Latex Export

```
[fontsize=]elisp (add-to-list 'safe-local-eval-forms '(add-hook 'after-save-hook 'org-latex-export-to-pdf t t))
```

5.10.3 Org-babel

```
[fontsize=]elisp (add-to-list 'safe-local-eval-forms '(add-hook 'after-save-hook 'org-babel-tangle t t))
```

5.11 format all error

```
[fontsize=]elisp (add-to-list '+format-on-save-enabled-modes 'web-mode 1)
```

5.12 gnuplot

This block changes the default term type for png file types [fontsize=]elisp (after! gnuplot (add-to-list '*org-babel-gnuplot-terms* '(png . "pngcairo transparent")))

5.13 org-fragtog

```
[fontsize=]elisp (use-package! org-fragtog :after org :hook (org-mode . org-fragtog-mode) :config)
```

5.14 ispell

5.15 Startup

```
[fontsize=]elisp (after! org (setq! org-startup-with-latex-preview t) (setq! org-startup-with-inline-images t) (setq! org-latex-image-default-width "1in") (setq! org-cite-export-processors '(t csl)) (setq! org-cite-global-bibliography (list" /Dropbox/org/citations/citations.json")) (setq! yas/triggers-in-field t) (setq! org-xournalpp-image-type 'png) (setq! org-export-allow-bind-keywords t)) (with-eval-after-load 'ox-latex (add-to-list 'org-latex-classes '("apa" "documentclass[11pt]apa7" ("part("
```

```

chapter("
section("
subsection("
subsubsection(with-eval-after-load 'ox-latex (add-to-list 'org-latex-classes '("mla" "
documentclassmla" ( "
part("
chapter("
section("
subsection("
subsubsection

```

5.16 citations

```

[fontsize=]elisp (use-package! bibtex-completion :defer t :config (setq bibtex-completion-additional-
search-fields '(keywords) bibtex-completion-pdf-field "file")) ; This tell bibtex-completion to look
at the File field of the bibtex to figure out which pdf to open
  (use-package! bibtex-actions :after embark bibtex-completion :config (add-to-list 'embark-
keymap-alist '(bibtex . bibtex-actions-map)))
  (use-package! citeproc :defer t)
  ;; Org-Cite configuration
  (use-package! oc :after org bibtex-completion bibtex-actions :config (require 'ox) (map! :map
org-mode-map :localleader :desc "Insert citation" "@ " 'org-cite-insert) (defvar bibtex-actions-
bibliography nil) (setq org-cite-global-bibliography (let ((paths (or bibtex-actions-bibliography
bibtex-completion-bibliography))) ;; Always return bibliography paths as list for org-cite. (if
(stringp paths) (list paths) paths))) ;; setup export processor; default csl/citeproc-el, with bibla-
tex for latex (setq org-cite-export-processors '((t csl))))
  ;; Org-cite processors
  ;;; Core
  (use-package! oc-basic :after oc)
  (use-package! oc-biblatex :after oc)
  (use-package! oc-csl :after oc :config (setq org-cite-csl-styles-dir " /Zotero/styles"))
  (use-package! oc-natbib :after oc)

```

5.17 set header args

```

[fontsize=]elisp (setq! org-global-properties '(("header-args:latex" . " :results output file graph-
ics :imagemagick yes :headers '
..
usepackagetikz

usepackagesunitx

usepackagegensymb) :fit yes :iminoptions -density 600) ("header-args" . " :pandoc t"))

```

5.18 org-xournalpp

```
[fontsize=]elisp ;;(use-package! org-xournalpp ;; :config ;; (add-hook 'org-mode-hook 'org-xournalpp-mode))
```

5.19 Disable Line wrapping

```
[fontsize=]elisp (after! org (setq! org-startup-truncated 'nil) )
```

5.20 Org pomodoro

```
[fontsize=]elisp (setq alert-user-configuration (quote (((:category . "org-pomodoro"))) libnotify nil))) (defun david/org-pomodoro-time () "Return the remaining pomodoro time" (if (fboundp 'org-pomodoro-active-p) (if (org-pomodoro-active-p) (cl-case org-pomodoro-state (:pomodoro (format "Pomo: (:short-break (format "Short break time: (:long-break (format "Long break time: (:overtime (format "Overtime! "No active pomo") "no active pomo")) (use-package org-pomodoro :ensure t :commands (org-pomodoro) :config (setq org-pomodoro-length 50 org-pomodoro-short-break-length 10 ))
```

5.21 nix hack

Org mode (latex export) has the wrong time. Not sure how to fix. This does not work.

5.22 plantuml

```
[fontsize=]elisp (after! org (setq! org-plantuml-exec-mode 'plantuml))
```

6 Latex setup

6.1 Use LuaTeX

```
[fontsize=]elisp (setq! TeX-engine 'luatex) (after! org ;(setq! org-latex-pdf-process '("PDFLATEX=luatex LATEX=luatex texi2dvi -pdf -clean -verbose -batch -shell-escape -output-directory=(setq! org-latex-pdf-process '("latexmk -f -pdf -(after! org (setq! org-latex-compiler "luatex")) (after! org (setq! org-latex-listings 'minted)) (after! org (setq! org-latex-minted-options '( "fontsize" "scriptsize" ) ) ) (after! org (add-to-list 'org-latex-classes '("factsheet" "documentclass[11pt]apa7" "part;;("chapter("section("
```

```
subsection("
subsubsection)
```

6.2 Keymap

```
[fontsize=]elisp (map! (:after auctext :map LaTeX-mode-map :leader :desc "compile" "c" 'TeX-
command-master))
```

7 Spell setup

7.1 Personal Dictionary

```
[fontsize=]elisp (setq! ispell-personal-dictionary " /.config/spell/dict.txt")
```

7.2 Fix hunspell bug

```
[fontsize=]elisp (setq ispell-program-name "hunspell") (ispell-check-version)
```

8 Python setup

8.1 Anaconda directory

```
[fontsize=]elisp (setq conda-anaconda-home " /opt/anaconda")
```

8.2 LSP nix

```
[fontsize=]elisp (after! lsp-python-ms (setq lsp-python-ms-executable (executable-find "python-
language-server")) (set-lsp-priority! 'mspyls 1)) (after! lsp-rust-rls (setq lsp-rust-rls-server-command
(executable-find "rls")) (set-lsp-priority! 'rls 1))

(after! lsp-clients-lua-language-server (setq lsp-clients-lua-language-server-bin (executable-
find "lua-language-server")) (set-lsp-priority! 'lua-language-server 1))
```

9 R setup

9.1 keymap

```
[fontsize=]elisp (map! (:after ess-mode :map ess-mode-map :nvi "C-<return>" 'ess-eval-line-
and-step :nvi "M-e" 'insert-R-assign ) )
```


9.2 Font lock keywords

9.2.1 R-major-mode

```
[fontsize=]elisp (setq ess-R-font-lock-keywords '( (ess-R-fl-keyword:keywords . t) (ess-R-fl-keyword:constants . t) (ess-R-fl-keyword:modifiers . t) (ess-R-fl-keyword:fun-defs . t) (ess-R-fl-keyword:assign-ops . t) (ess-R-fl-keyword:(ess-fl-keyword:fun-calls . t) (ess-fl-keyword:numbers . t) (ess-fl-keyword:operators . t) (ess-fl-keyword:delimiters . t) (ess-fl-keyword:= . t) (ess-R-fl-keyword:FT . t) ) ) )
```

9.2.2 R-inferior-mode

```
[fontsize=]elisp (setq inferior-ess-r-font-lock-keywords '( (ess-R-fl-keyword:keywords . t) (ess-R-fl-keyword:constants . t) (ess-R-fl-keyword:modifiers . t) (ess-R-fl-keyword:fun-defs . t) (ess-R-fl-keyword:assign-ops . t) (ess-R-fl-keyword:(ess-fl-keyword:fun-calls . t) (ess-fl-keyword:numbers . t) (ess-fl-keyword:operators . t) (ess-fl-keyword:delimiters . t) (ess-fl-keyword:= . t) (ess-R-fl-keyword:FT . t) ) ) )
```

9.3 Custom Functions

```
[fontsize=]elisp (defun insert-R-assign () "Inserts the assign statement in R <-" (interactive) (insert "<-" ) )
```

10 General configuration

10.1 remove line numbers

```
[fontsize=]elisp (setq display-line-numbers-type nil)
```

10.2 Fix treemacs ace-window bug

```
[fontsize=]elisp (require 'ace-window)
```

10.3 deletes compilation buffer if successful (ignores python buffers)

```
[fontsize=]elisp (add-hook 'compilation-finish-functions (lambda(buffer string) (if (and (null (string-match ".*exited abnormally.*" string)) (null(eq major-mode 'inferior-python-mode))) ;; make compilation window go away after a few seconds (progn (run-at-time "1 sec" nil 'delete-windows-on (get-buffer-create "**compilation**")) (print major-mode) (message "Compilation finished successfully")))))
```

10.4 ligatures

```
[fontsize=]elisp (setq +ligatures-extras-in-modes '(org-mode))
```

10.5 ispell dictionary

```
[fontsize=]elisp (setq! ispell-dictionary "en_US")
```

11 Key Map

11.1 General Buffers

```
[fontsize=]elisp (map! :leader :desc "treemacs" "0" 'treemacs :desc "last-buffer" "l" 'evil-switch-to-windows-last-buffer :nv "" nil (:prefix ("w") :desc "ace-window" "a" 'ace-window))
```

11.2 Python mode map

```
[fontsize=]elisp (map! (:map python-mode-map :localleader :desc "repl" "" '+python/open-python-repl (:prefix ("s" . "send") :desc "buffer" "b" 'python-shell-send-buffer :desc "function" "f" 'python-shell-send-defun :desc "region" "r" 'python-shell-send-region :desc "statement" "s" 'python-shell-send-statement)))
```

11.3 prolog map

```
[fontsize=]elisp (map! (:map prolog-mode-map :localleader :desc "repl" "" 'run-prolog :desc "file" "f" 'prolog-consult-buffer :desc "region" "r" 'prolog-consult-region :desc "predicate" "p" 'prolog-consult-region))
```

11.4 Doc-view mode map

```
[fontsize=]elisp (map! (:map doc-view-mode-map :nv "l" 'doc-view-next-page :nv "h" 'doc-view-previous-page))
```

11.5 mips mode map

```
[fontsize=]elisp (map! (:map mips-mode-map :localleader (:prefix ("s" . "send") :desc "file" "f" 'mips-run-file :desc "region" "r" 'mips-run-region :desc "buffer" "b" 'mips-run-region)))
```

11.6 haskell map

```
[fontsize=]elisp (map! (:map haskell-mode-map :localleader :desc "send-file" "f" 'haskell-process-load-file :desc "open-haskell" "" 'run-haskell))
```

12 remote

```
[fontsize=]elisp (setq projectile-file-exists-remote-cache-expire nil) (setq tramp-auto-save-directory  
" /Documents/tramp-autosave")
```

13 Helpful info

13.1 Font variables

- doom-font (normal font)
- doom-variable-pitch-font (easy reading font)
- doom-big-font (doom-big-font-mode for presentations)

13.2 Useful customization functions

- load! (load external .el files)
- use-package! (for configuring packages)
- after! (runs config after packages has loaded)
- add-load-path! (adds directories to load-path variable)
- map! (binds keys)