↖ Topic Index

Writing Tools 🚧

Operation	Keystroke	Function	Note
Writing Tools ○ Help & Customize • artbollocks-mode • writegood-mode • wc-mode (counts & goals) • writeroom-mode ○ **my-whisper**	Several external packages provide supplemental writing tools to what Emacs already provides. See also: **Σ Spell Checking** PEL installs the following external packages when their corresponding pel-use- customizable user-option is turned on (set to **t**):		
	📦 **artbollocks-mode**	🔄 **pel-use-artbollocks-mode**	A minor mode to avoid clichés and bad grammar
	📦 **gt**	🔄 **pel-use-go-translate**	Text translator, supports multiple engines: google, Bing, deepL, StarDict, Youdao & LLMs ChatGPT, DeepSeek, etc..

	📦 **harper-ls**	🔄 **pel-use-harper-ls**	A minor-mode to use Harper, a locally hosted, privacy-first, fast, open-source grammar checker.
			☝ You must manually **install the harper-ls server** on your system.
			• Set pel-use-harper-ls user-option to either:
			  • A list of major modes where it must be activated
			  • The value **english-prose-mode-plaintext**

	📦 **my-whisper**	🔄 **pel-use-my-whisper**	Speech-to-text via the **Whisper.cpp** port of OpenAI Whisper model. All audio is processed **locally**. ☝ You must manually **install Whisper.cpp and other dependencies** on your system.
	📦 **wc-mode**	🔄 **pel-use-wc-mode**	The wc-mode minor mode counts words with optional word-count goal.
	📦 **writegood-mode**	🔄 **pel-use-writegood-mode**	A minor mode to aid in finding common writing problems.
	📦 **writeroom-mode**	🔄 **pel-use-writeroom-mode**	A minor-mode that implements distraction-free writing mode.
Last updated on:	2025-11-29	To activate above package in PEL access the **pel-pkg-for-writing** customization group. Access it with:	`<f11> W <f2>`
Open this PDF file. See also: **Σ Help/Info**	`<f11> W <f1>`	(**pel-help-pdf** &optional OPEN-WEB-PAGE)	Open the **Writing Tools** local PDF. If the prefix argument (like `C-u` or `M--`) is used, then it opens the remote GitHub hosted raw PDF instead. If the **pel-flip-help-pdf-arg** user-option is set it's the other way around.
Σ Customize PEL Writing Tools support	`<f11> W <f2>`	(**pel-customize-pel** &optional OTHER-WINDOW)	Customize PEL Writing Tools support: open **pel-pkg-for-writing-mode** group giving access to all **pel-use-** user options listed above. • If OTHER-WINDOW is non-nil (use `C-u`), display in another window.
Σ Customize Emacs Writing Tools support	`<f11> W <f3>`	(**pel-customize-library** &optional OTHER-WINDOW)	Customize Writing Tools external packages support: artbollocks-mode, wc, writegood, writeroom. • If OTHER-WINDOW is non-nil (use `C-u`), display in another window.
Toggle artbollocks-mode to highlight various English prose issues	`<f11> W a`	(**artbollocks-mode** &optional ARG)	Highlight passive voice, weasel words and artbollocks jargon in text, and provide useful text metrics. 📦 Requires **artbollocks-mode** 🔄 activated by **pel-use-artbollocks-mode.** ☝ In some cases you might need to force re-fontification win the buffer, by executing: `M-x font-lock-fontify-buffer`
Count words in the marked area	`C-c [`	(**artbollocks-word-count** &optional START END)	Count the number of words in the marked area or complete buffer and print it in the echo area.
Count sentences in the marked area	`C-c ]`	(**artbollocks-sentence-count** &optional START END)	Count the number of sentences in the marked area or complete buffer and print it in the echo area.
Evaluate the readability index of marked section.	`C-c \`	(**artbollocks-readability-index** &optional START END)	Determine the automated readability index of the text or complete buffer in the marked area.
Evaluate the Flesch reading ease of text in marked section.	`C-c /`	(**artbollocks-reading-ease** &optional START END)	Determine the Flesch reading ease of text in the marked area or complete buffer, print the rating in the echo area. • Higher scores indicate material that is easier to read.
Evaluate and print Flesch-Kinkaid grade level of text in the marked	`C-c =`	(**artbollocks-grade-level** &optional START END)	Determine the Flesch-Kinkaid grade level of text in the marked area and print it in the echo area.
Toggle writegood-mode to highlight various English prose issues	`<f11> W g g`	(**writegood-mode** &optional ARG)	Colorize issues with the writing in the buffer. 📦 Requires **writegood-mode** 🔄 activated by **pel-use-writegood-mode**
Print writegood-mode version	`<f11> W g v`	(**writegood-version**)	Print the version of write good-mode.el
Evaluate the Flesch reading ease of text in marked section.	`<f11> W g r`	(**writegood-reading-ease** &optional START END)	Print Flesch reading ease and Flesch-Kinkaid grade level of text in the marked area or complete buffer. Scores roughly between 0 and 100.
Evaluate and print Flesch-Kinkaid grade level of text in the marked	`<f11> W g l`	(**writegood-grade-level** &optional START END)	Flesch-Kincaid grade level test. Converts reading ease score to a grade level (Score ~ years of school needed to read passage).
Toggle wc-mode (Word Count Mode)	`<f11> W c`	(**wc-mode** &optional ARG)	Toggle wc mode (Word Count mode). • When wc-mode is enabled on a buffer, it counts the current words in the buffer and keeps track of a differential of added or subtracted words. • A goal of number of words added/subtracted can be set while using this mode. Upon completion of the goal, the modeline text will highlight indicating that the goal has been reached.
Set word goal	`C-c C-w w`	(**wc-set-word-goal** GOAL)	Set a goal for adding or removing words in the buffer.
Set line goal	`C-c C-w l`	(**wc-set-line-goal** GOAL)	Set a goal for adding or removing lines in the buffer.
Set character goal	`C-c C-w a`	(**wc-set-char-goal** GOAL)	Set a goal for adding or removing chars in the buffer.
Count words	`C-c C-w c`	(**wc-count** &optional RSTART REND FIELD)	Count the words, lines and characters present in the region following point. This function follows most of the rules present in the 'how-many' function. If INTERACTIVE is omitted or nil, just return the word count, do not print it. Otherwise, if INTERACTIVE is t, the function behaves according to interactive behavior. • START and END specify the region to operate on. • When called interactively, this function first checks to see if it is in Transient Mark mode. If that is the case, then the function operates over the marked region. Otherwise, it will operate over the entire buffer.
Toggle writeroom-mode	`<f11> W r r`	(**writeroom-mode** &optional ARG)	Minor mode for distraction-free writing. 📦 Requires **writeroom-mode** 🔄 activated by **pel-use-writeroom-mode**
Increase width	`<f11> W r +`	(**writeroom-increase-width**)	Increase the width of the writing area by 2 characters.
Decrease width	`<f11> W r -`	(**writeroom-decrease-width**)	Decrease the width of the writing area by 2 characters.
Adjust width	`<f11> W r 0`	(**writeroom-adjust-width** AMOUNT)	Adjust the width of the writing area on the fly by AMOUNT. • A numeric prefix argument can be used to specify the adjustment. • When called without a prefix, this will reset the width to the default value
Toggle display of mode line	`<f11> W r m`	(**writeroom-toggle-mode-line**)	Toggle display of the mode line (shows modeline at the top of buffer). ☝ Since this command is allowed while the wriiteroom-mode is off, you can use it to add a modeline bar at the top of the current window. That window will then have **2** mode lines: one at the top and one at the bottom.

Operation	Keystroke	Function	Note
Toggle my-whisper mode **Transcribe speech to text using a AI LLM running locally on your computer.**	`<f11> W W`	(**my-whisper-mode** &optional *ARG*)	Toggle the my-whisper global minor mode on/off. • When turned on it starts recording audio in the background. After some audio is recorded you can stop the recording by turning the mode off or using the **my-whisper-stop-record** command. Stopping audio recording starts the speech to text transcribing process and the transcribed text is inserted in the current buffer at point.
	📦 Requires **my-whisper** 🤖 activated by **pel-use-my-whisper** • This also requires the SoX library and the Whisper.cpp project. See my-whisper.cpp home page for information to get and build those. • The AI LLM model used for the analysis is selected by the **my-whisper-model** user-option. Several models are provided in the whisper.cpp/model directory that you need to install and build separately.. More models are available at ggerganov/whisper.cpp @ Hugging Face. When the mode is active, one of its mode lighters shows inside the mode line of every window: • When recording, the mode line is: " 🎙️ " Change it via the **my-whisper-lighter-when-recording** user-option. • When no recording, the mode line is: " ⏸️ " Change it via the **my-whisper-lighter-when-idle** user-option. ☝️ If you just want to record, stop and insert transcribed text and do nothing else, then: • start recording with `<f11> W W` • stop recording with `<f11> W W` (or `<f5>` to repeat last command if you did not type anything since you started recording). • Two key bindings are made available while the minor mode is active. See below. 🎛️ The two commands made available while **my-whisper-mode** is active are listed below with their default key bindings. You can change these key bindings via customization. See below.		
Stop audio recording, transcribe text at point.	`C-c .`	(**my-whisper-stop-record**)	Stop recording. Insert transcribed text in current buffer at point. 🎛️ The key binding can be modified in the **my-whisper-key-for-stop-record** user option.
Start recording again.	`C-c ,`	(**my-whisper-record-again**)	Start recording again. 🎛️ The key binding can be modified in the **my-whisper-key-for-record-again** user option.
Transcribe an already recorded WAV file	`<f11> W w`	(**my-whisper-transcribe-file** FNAME)	Transcribe a recorded audio file FNAME, insert transcribed text at point. • ☝️This command can only be used when ‘my-whisper-mode is inactive.