# GNU Introduction

You are reading about GNU Emacs, the GNU incarnation of the advanced, self-documenting, customizable, extensible editor Emacs. (The ‘G’ in GNU (GNU’s Not Unix) is not silent.)

We call Emacs “advanced” because it can do much more than simple insertion and deletion of text. It can control subprocesses, indent programs automatically, show multiple files at once, and more. Emacs editing commands operate in terms of characters, words, lines, sentences, paragraphs, and pages, as well as expressions and comments in various programming languages.

“Self-documenting” means that at any time you can use special commands, known as “help commands”, to find out what your options are, or to find out what any command does, or to find all the commands that pertain to a given topic. \*Note Help.

“Customizable” means that you can easily alter the behavior of Emacs commands in simple ways. For instance, if you use a programming language in which comments start with ‘<\*\*’ and end with ‘\*\*>’, you can tell the Emacs comment manipulation commands to use those strings (\*noteComments::). To take another example, you can rebind the basic cursor motion commands (up, down, left and right) to any keys on the keyboard that you find comfortable. \*Note Customization[[1]](#footnote-21).

This is so according to Clear (2019). Ahrens describes this as the main objective of note-taking (2017).

Equation 1.[[2]](#footnote-22)

“Extensible” means that you can go beyond simple customization and create entirely new commands. New commands are simply programs written in the Lisp language, which are run by Emacs’s own Lisp interpreter. Existing commands can even be redefined in the middle of an editing session, without having to restart Emacs. Most of the editing commands in Emacs are written in Lisp; the few exceptions could have been written in Lisp but use C instead for efficiency. Writing an extension is programming, but non-programmers can use it afterwards.

\*Note Emacs Lisp Intro: (eintr)Top, if you want to learn Emacs Lisp programming.

# Bibliography

Ahrens, Sönke. 2017. *How to Take Smart Notes: One Simple Technique to Boost Writing, Learning and Thinking – for Students, Academics and Nonfiction Book Writers*. Kindle Edition. Amazon.com Services LLC.

Clear, David B. 2019. “Zettelkasten — How One German Scholar Was so Freakishly Productive.” Medium. December 31, 2019. <https://writingcooperative.com/zettelkasten-how-one-german-scholar-was-so-freakishly-productive-997e4e0ca125>.

1. See section on Customization. [↑](#footnote-ref-21)
2. Source: <https://en.wikipedia.org/wiki/Matrix_(mathematics)> [↑](#footnote-ref-22)