PEL Topics Index

Last updated on: 2025-09-01 — Note: with PEL, type `<f11> <f1>` to open this PDF index.

Emacs Reference Cards

☝ With PEL, access these cards from Emacs with the `<f11> ? e r` key sequence.
See ⅀ **Help/Info** for more info.

These are links to the PDF version of official English version of the quick reference cards for **GNU Emacs** and popular external packages. PEL documents Emacs key bindings as well, these cards provide useful complement to what PEL provides.

Emacs	**Calc**	**Gnus**	**Magit Cheatsheet**	**Org**	**Viper**
Emacs survival card	**Dired**	**Gnus booklet**	**Magit Ref-card**		**VIP**

➢ PEL Overview

- **PEL repo**
- **PEL Readme**
- **PEL Manual**
- **PEL NEWS** 🗞
- **Discussions**

PEL license

Last updated on: 2025-09-01

Emacs Mailing Lists

This table holds links to the **PEL file tables** (hosted on Github as raw PDF files).

☝ For the best user experience, use a browser that can render PDF directly instead of downloading.

- **Mozilla Firefox** (version > 78) does that perfectly. You may need to activate a plug-in for other browsers.
- With that in place, you can browse through all the PDFs and reach a vast amount of information quickly.

☝ From within Emacs open this topic index PDF by typing the `<f11> ? <f1>` key sequence. More help topics with `<f11> ? p` keys.

☝ The symbols, **colour coding** and various other conventions are described in the **➢Legend** PDF.

Terminal Multiplexers: GNU screen , Tmux
Command Line Scripting Languages: bash, sh, zsh
Cmdline: GNU readline, ls -l

General Info ➢	➢Legend	➢Recommended Emacs User Option		➢Themes	Migrate from CRiSP	
Startup ➢		Run Emacs daemon & clients 🐧		🖥iMenu/Speedbar support		
PEL Code ➢	How to do it with PEL	🖥PEL Naming Conventions		🖥PEL Environment Variables		🖥PEL utilities

OS Desktop Key Bindings ⌨ (Bindings that don't clash with PEL)	macOS Fct Keys	macOS Keys	🐧Mint 20 Desktop Keys	🐧Ubuntu 16.04 Desktop Keys
		terminal settings	🐧Rocky Linux 8 Desktop Keys	

🚦 **Feature Comparisons**	🚦 Completion Modes Compatibility		🚦 Speedbar/iMenu Mode Compatibility		🚦 Shells/Terminals Comparisons	
Key Prefixes & Suffixes	⅀ ⌨Modifier Keys	⅀⌨Numkeypad	⌨Keys - Fn	⌨Keys - F11	⌨Keys - F12	➢PEL

⅀ **Emacs Manual** , **Guided Tour of Emacs.**

- **Mastering Emacs** , **Awesome-Emacs**
- **MELPA** and **GNU ELPA**

The tables listed at right describe Emacs commands & key bindings for concepts & features. The cell is light-blue for major mode, light-red for minor mode specific concepts. Grey cells are links into other pages for important concepts.
Emacs commands can be executed by name or bound to key sequences. They describe the commands, their arguments and the key sequences bound to them.

- **Emacs Keys**
- **Numeric Arguments**

You can also:

- **Run Command by Name**

Emacs uses a concept of modes:

- **Emacs Major and Minor Modes**
  - **Major Modes**
  - **Minor Modes**
  - **Choosing Modes**

PEL provides several key sequences to toggle minor modes.

Cells link titles starting with only ⅀ are Emacs generic features, blue links are external packages. The green links are mostly PEL extensions.

⅀ Abbreviations	⅀ Diff & Merge	⅀ Grep	⅀ Marking	⅀ Scrolling	⅀ Tab Bar
⅀ Align	⅀ Dired	⅀ Help/Info	⅀ Menus	⅀ Search/Replace	T Templates
⅀ Auto-Completion	⅀ Display - Lines	⅀ Hide/Show	⅀ Mode Line	⅀ Sessions	⅀ Text Modes
⅀ Autosave/Backup	⅀ Drawing	⅀ Highlight (colors)	⅀ Mouse	⅀ start Shells/REPLs	⅀ Time Tracking
⅀ Bookmarks	⅀ Enriched Text	⅀ ibuffer-mode	⅀ Narrowing	⅀ shell-mode	⅀ Tramp 🛜
⅀ Buffers	⅀ Execute Cmds	⅀ Indentation	⅀ Navigation	⅀ term-mode	⅀ Transpose text
⅀ Case Conversions	⅀ Exec Shell Cmds	⅀ Input Method	⅀ Object Files	⅀ eat-mode	⅀𝔛 Treemacs
⅀ Close/Suspend	⅀ Faces/Fonts	⅀ Inserting Text	⅀ Outline	⅀ vterm-mode	⅀ Undo/Redo/Repeat
⅀ Comments	⅀P Fast Startup	⅀ Key-Chords	⅀ Packages	⅀𝔛 Smartparens	⅀ VCS-Git 𝔛Magit
⅀ Compilation Mode	⅀ File Encoding	⅀ Keyboard Macros	⅀𝔛 Projectile	⅀ Sorting	⅀ VCS-Mercurial
⅀ Completion/Input	⅀ File-mngt	𝔓𝔩𝔛- Lispy	⅀ Recursive Edit	⅀ Speedbar	⅀ VCS-Subversion
⅀ Counting	⅀ File/Dir Variables		⅀ Rectangles	⅀ Spell Checking	⅀ Web
⅀Ɱ CUA	⅀ Fill/Justify		⅀ Registers	⅀ SyntaxCheck	⅀ Whitespace
⅀ Cursor	⅀ Frames				⅀ Windows
⅀ Customize					⅀ Xref - Cross Refs
⅀ Cut & Paste					

ℒ𝔓𝔩 - **Emacs Lisp** concepts & tools	ℒ display-buffer	ℒ※ - ELisp Types	ℒ ERT (regr-testing)	ℒ Hooks		

XRef - Cross Reference Tools See also: ⅀ **Xref**

Emacs supports various cross reference mechanisms described in the ⅀ **Xref** table. These mechanisms take advantage of various external tools and integrate with them. Notes about those tools are available in the tables listed in this section. Also describes indentation.

🚦 Xref-Support	🚦 Xref-Frontend	🚦 Xref-Backend			🚦 Indentation Styles

Build Tools & Preprocessor	𝔓𝔩 - CMake 🚧	𝔓𝔩 - M4	𝔓𝔩 - Make gmake	𝔓𝔩 - Nix	𝔓𝔩 - Tup	
Data Serialization & Modelling	Ⓓ CWL	Ⓓ YAML		Ⓢ ASN.1 asn1-mode	Ⓢ MIB snmp-mode	Ⓢ YANG
Other File Formats	⅀ Changelog Files	Config/ini/toml... Files	RFC (RFC @ Wikipedia)	RPM Files 🐧 (spec file format)		Ɱ X.509 Certificates
Hardware Description Languages	Verilog 🚧	VHDL 🚧				
Lightweight Markup Languages	Ɱ AsciiDoc	Ɱ Markdown	Ɱ Org-Mode	Ɱ reStructuredText		
• **Graphics Markup**	Ɱ Graphviz Dot	Ɱ MscGen	Ɱ PlantUML			

Programming Languages
Main Paradigm of Programming Languages

- *Actor Model:* Ⓐ *Array* ☒
- *Concatenative* Ⓚ *Concurrent:* ⓒ
- *Domain Specific* ⓓ
- *Dynamic* 𝒹 *Extensible* ⓔ
- *Functional:* ⓕ *Pure:* Ⓕ
- *Generic* ⓖ
- *Imperative:* ⓘ *or no token*
- *Object Oriented* ⓞ *Procedural* ⓟ
- *Has Syntactic Macros:* ⓜ
- *Multi-paradigm* ⤳ *Reflective* |
- *System Level* 🅢

- The programming languages supported by PEL are listed here in alphabetical order.
- Emacs (and PEL) also provides basic support for some of the one PEL does not support and for other programming languages not listed here.

Future support for APL, Carbon, Crystal, Dart, Elm, Groovy, Haxe, Kotlin, Purescript, ReasonML, Scala, Typescript and documentation of support for Fortran, Javascript, Java, Modula, (based on my need for them or requests).

Emacs has major mode support for several programming languages. PEL extends Emacs support for some of them (others are marked 🚧).

BEAM Programming	*Functional*	Javascript target	Pascal-style syntax	Lisp-like Languages	Stack Based
Curly Bracket	Java Virtual Machine	ML Family	Lisp Family	Scheme Dialects	OS App Control

𝔓𝔩 - Ada 🚧 ⤳🅢	𝔓𝔩 - D ⓘⓕⒶ	𝔓𝔩 - Gambit ⓕⓜ	𝔓𝔩 - Janet ⓘⓕⓜ	𝔓𝔩-Pascal	Scala 🚧
𝔓𝔩- AppleScript	Dart 🚧	𝔓𝔩 - Gerbil ⓕⓜⒶ	Java 🚧	𝔓𝔩 - Perl (perl5)	𝔓𝔩 - Scheme ⓕⓜ
APL 🚧	𝔓𝔩 - Eiffel 🚧 ⓞ🅢	𝔓𝔩 - GNU Guile ⓕⓜ	𝔓𝔩 - Javascript 🚧	𝔓𝔩 - Pike 𝒹 ⓘ ⓞ	𝔓𝔩-Seed7 🚧 ⓔ ⓖ ⤳
𝔓𝔩 - Arc ⓕⓜ	𝔓𝔩 - Elm 🚧 Ⓕ	𝔓𝔩 - Gleam	𝔓𝔩 - Julia ⓜ	𝔓𝔩 - *Python* 𝒹ⓟⓞⓕ	𝔓𝔩-Smalltalk 🚧 ⓞ
𝔓𝔩 - awk ⓓ	𝔓𝔩 - Elixir ⓒⓜⓕⒶ	𝔓𝔩 - Go 🅢	Kotlin 🚧	𝔓𝔩 - Purescript 🚧 Ⓕ	𝔓𝔩-Swift
𝔓𝔩 - C 🅢	ℒ𝔓𝔩 - Emacs Lisp	Groovy 🚧	𝔓𝔩 - LFE ⓒⓜⓕⒶ	R 🚧 ⓞ ⓟ ⓕ ☒ |	𝔓𝔩 - Tcl ⓕⓘ
𝔓𝔩 - C++ ⓞ🅢	𝔓𝔩 - Erlang ⓒⓕⒶ	𝔓𝔩 - Haskell Ⓕ	𝔓𝔩 -Lua ⓕ ⓞ ⓟ	𝔓𝔩 - Racket ⓕⓜ	𝔓𝔩 - Typescript 🚧
Carbon 🚧 future 🅢	𝔓𝔩 - Factor Ⓚⓕⓞⓜ	Haxe 🚧	𝔓𝔩-Modula	𝔓𝔩 - ReasonML 🚧	𝔓𝔩 - UNIX Shell
𝔓𝔩 - Chez ⓕⓜ	𝔓𝔩 - Forth Ⓚ	𝔓𝔩 - Hy (python) ⓜ	𝔓𝔩 - NetRexx	𝔓𝔩 - REXX	𝔓𝔩 - V
𝔓𝔩 - Chibi ⓕⓜ	Fortran 🚧		𝔓𝔩 - Nim ⓜ🅢	𝔓𝔩 - Ruby	𝔓𝔩 -Zig 🅢
𝔓𝔩 - Chicken ⓕⓜ			𝔓𝔩-Objective-C 🚧	𝔓𝔩 - Rust 🅢	
𝔓𝔩 - Clojure ⓕⓜ			𝔓𝔩 - OCaml ⓘⓕ		
Common Lisp ⓕⓜ			𝔓𝔩 - Odin 🅢		
Crystal 🚧					