PEL Topics Index

Last updated on:	2024-11-14	Note: with PEL, type `<f11> <f1>` to open this PDF index.	

Emacs Reference Cards

☝ With PEL you can access these via the `<f11> ? e r` key sequence.
See Σ **Help/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 (license)
- PEL repo
- PEL Readme
- PEL Manual
- PEL NEWS 📝
- Discussions

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**

General Info. ➤
Startup. ➤
Development Info. ➤

➤Legend	➤Recommended Emacs User Option	➤Themes	Migrate from CRiSP
	Run Emacs daemon & clients 🍎🐧	🗔iMenu/Speedbar support	
	🗔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

Σ 🗔Modifier Keys	Σ🖮Numkeypad	🖮Keys - Fn	🖮Keys - F11	➤PEL

Σ Emacs Features
- A **Guided Tour of 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. Emacs commands can be executed by name or bound to key sequences. They describe the commands, their <u>arguments</u> 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 Σ are Emacs generic features, blue links are external packages. The green links are mostly PEL extensions.

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

𝕃𝓟𝕃 - Emacs Lisp concepts & tools	𝓛 display-buffer	𝓛✲ - ELisp Types	𝓛 ERT (regr-testing)	𝓛 Hooks

XRef - Cross Reference Tools See also: Σ Xref

Emacs supports various cross reference mechanisms described in the Σ 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.

🚦 Xref-Support	🚦 Xref-Frontend	🚦 Xref-Backend		

Build Tools & Preprocessor

PEL supports installation and partial setup of the following tools:

PEL has support for several build tools but they are not all documented in a page.
- **Nix** 📦 Requires **nix-mode** external package ☑ activated when **pel-use-nix-mode** user-option is tuned on.
- **Tup** 📦 Requires **tup-mode** external package ☑ activated when **pel-use-tup** user-option is tuned on.

Command Line Scripting Languages: bash, sh, zsh

𝕃𝓛 - CMake 🏗	𝕃𝓛 - M4	𝕃𝓛 - Make gmake	

Data Serialization

Ⓓ CWL	Ⓓ YAML		Utility: **GNU readline**

Data Modelling/ Specification

Ⓢ ASN.1 asn1-mode	Ⓢ MIB snmp-mode	Ⓢ YANG	ls -l

Other File Formats

	M RPM Files 🐧	M X.509 Certificates	

Hardware Description Languages

Verilog 🏗	VHDL 🏗		

Lightweight Markup Languages

M AsciiDoc	M Markdown	M Org-Mode	M reStructuredText	**OS App Control Scripting Languages**

• Graphics Markup

M Graphviz Dot	M MscGen	M PlantUML		𝕃𝓛🍎- AppleScript

Programming Languages
Main Paradigm of Programming Language Families
- *Actor Model:* Ⓐ
- *Concatenative:* Ⓚ
- *Concurrent:* Ⓒ
- *Domain Specific:* ⓓ
- *Dynamic:* 𝒹
- *Functional:* ⓕ *Pure:* Ⓕ
- *Imperative:* ⓘ or no token
- *Object Oriented:* ⓞ
- *Procedural:* Ⓟ
- *Has Syntactic Macros:* ⓜ
- *System Level:* Ⓢ

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

Future support for Crystal, Elm, Kotlin, Lua, Purescript, ReasonML, Seed7, Typescript, Zig and documentation of support for Ada, Fortran, Javascript, Java, Modula, Pascal (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 Languages	*Functional Languages*	Javascript target	Lisp Family Languages	Lisp-like Languages	
Curly Bracket Languages	Java Virtual Machine Languages	ML Family Languages	Scheme Language Dialects	Stack Based Languages	

Cell colours identifies the programming language family(ies).

Ada 🏗	𝕃𝓛 - D ⓘⓕⒶ	𝕃𝓛 - Gambit ⓕⓜ	𝕃𝓛 - Janet ⓘⓕⓜ	Objective-C 🏗	Scala 🏗
𝕃𝓛 - Arc ⓕⓜ	Dart 🏗	𝕃𝓛 - Gerbil ⓕⓜⒶ	Java	𝕃𝓛 - OCaml ⓘⓕ	𝕃𝓛 - Scheme ⓕⓜ
𝕃𝓛 - awk ⓓ	Eiffel 🏗 Ⓢ	𝕃𝓛 - GNU Guile ⓕⓜ	𝕃𝓛 - Javascript 🏗	Pascal 🏗	Seed7 🏗
𝕃𝓛 - C Ⓢ	𝕃𝓛 - Elm 🏗 Ⓕ	𝕃𝓛 - Gleam	Σ Julia ⓜ	𝕃𝓛 - Perl	Swift 🏗
𝕃𝓛 - C++ ⓞⓈ	𝕃𝓛 - Elixir ⓒⓜⓕⒶ	𝕃𝓛 - Go Ⓢ	Kotlin 🏗	𝕃𝓛 - Python 𝒹Ⓟⓞⓕ	𝕃𝓛 - Tcl 🏗 ⓕⓘ
𝕃𝓛 - Chez ⓕⓜ	𝕃𝓟𝕃 - Emacs Lisp	Groovy 🏗	𝕃𝓛 - LFE ⓒⓜⓕⒶ	𝕃𝓛 - Purescript 🏗 Ⓕ	𝕃𝓛 - Typescript 🏗
𝕃𝓛 - Chibi ⓕⓜ	𝕃𝓛 - Erlang ⓒⓕⒶ	𝕃𝓛 - Haskell Ⓕ	Lua 🏗	𝕃𝓛 - Racket ⓕⓜ	𝕃𝓛 - UNIX Shell
𝕃𝓛 - Chicken ⓕⓜ	Factor Ⓚⓕ ⓞⓜ	Haxe 🏗	Modula 🏗	𝕃𝓛 - ReasonML 🏗	𝕃𝓛 - V
𝕃𝓛 - Clojure ⓕⓜ	𝕃𝓛 - Forth Ⓚ	𝕃𝓛 - Hy (python) ⓜ	𝕃𝓛 - NetRexx	𝕃𝓛 - REXX	Zig 🏗 Ⓢ
Common Lisp ⓕⓜ	Fortran 🏗		𝕃𝓛 - Nim ⓜⓈ	𝕃𝓛 - Ruby	
Crystal 🏗				𝕃𝓛 - Rust Ⓢ	