

PEL Quick Access Topics Index

© Pierre Rouleau 2021-2026		Last updated on: 2026-07-09		Sections Help		Inside Emacs with PEL; type <f11> <f1> to open this PDF index.					
GNU Emacs Reference Cards - Emacs Release History - EmacsWiki - Emacs project repo		With PEL, access these PDF cards from within Emacs with the <f11> ? e r key sequence. See ℥ Help/Info for more info.									
		Emacs	Calc	Gnus	Magit Cheatsheet	Org	Viper				
		Emacs survival card	Dired	Gnus booklet	Magit Ref-card		VIP				
➤ PEL Repo Readme - License Manual - NEWS Discussions		This table holds links to all other PEL topic oriented PDF table files (hosted on Github Pages). 🙌 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 🔑: GNU readline, ls -l, ssh		General Info ➤	➤Legend		➤Recommended Emacs User Option		➤Themes	Migrate from CRiSP	℥✖ - Emacs startup		
		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					
Prefix/Suffix & Numeric Arg Keys		℥ ≡ Modifier Keys		℥ ≡ Numkeypad		≡ Keys - Fn		≡ Keys - F11			
								≡ Keys - F12			
								➤ PEL			
℥ Emacs Features ℥ Emacs Manual , Guided Tour of Emacs , Emacs Lisp Manual - Emacs Docs: Emacs, Emacs Lisp - Mastering Emacs, Awesome-Emacs - MELPA and GNU ELPA The tables at right describe Emacs concepts/features commands & key bindings. Cell background is light-blue for major mode, light-red for minor mode specifics, grey for links to sections of other tables. Cells link titles starting with ℥ are Emacs generic features, blue links are external packages. The green links are mostly PEL extensions. 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 and 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. <i>1979 EMACS Intro memo by R.M. Stallman</i>		℥ Abbreviations Debuggers🔑🔑		℥ Grep (grep-regexp)		℥ Man pages		℥ Scrolling		℥ Tab Bar	
		℥ Align		℥ Diff & Merge		℥ Help/Info		℥ Marking		℥ Search/Replace	℥ Templates
		℥ Auto-Completion		℥ Dired		℥ Hide/Show		℥ Menus ℥iMenu		℥ Sessions	℥ Text Modes
		℥ Autosave/Backup		℥ Display - Lines		℥ Highlight (colors)		℥ Mode Line		℥ start Shells/REPLs	℥ Time Stamps
		℥ Bookmarks		℥ Drawing		℥ ibuffer-mode		℥ Mouse		℥ shell-mode	℥ Time Tracking
		℥ Buffers		℥ Eldoc		℥ Indentation		℥ Narrowing		℥ term-mode	℥ Tramp 🗨️
		℥ Case Conversions		℥ Enriched Text		℥ Input Method		℥ Navigation		eat-mode	℥ Transpose text
		℥ Close/Suspend		℥ Execute Cmds		℥ Inserting Text		℥ Object Files		vterm-mode	℥ X Treemacs
		℥ Comments		℥ Exec Shell Cmds		℥ Key-Chords		℥ Outline		℥ X Smartparens	℥ Tree Sitter
		℥ Compilation Mode		℥ Faces/Fonts		℥ Keyboard Macros		℥ Packages		℥ Sorting	℥ Undo/Redo/Repeat
		℥ Completion/Input		℥ P Fast Startup		℥ Line Ending Mngt		Programming		Speech To Text	℥ VCS-Git xMagit
		℥ Copyright		℥ File Encoding		℥ X- Lispy		℥ Project Tools		℥ Speedbar	℥ VCS-Mercurial
		℥ Counting		℥ File-mngt		Logging key strokes		℥ X Projectile		℥ Spell Checking	℥ VCS-Subversion
		℥ M CUA		℥ File/Dir Variables				℥ Recursive Edit		℥ SyntaxCheck	℥ Web
		℥ Cursor		℥ Fill/Justify				℥ Rectangles			℥ Whitespace
		℥ Customize		℥ Frames				℥ Registers			℥ Windows
		℥ Cut & Paste									Writing Tools
											℥ Xref - Cross Refs
Emacs Lisp Ref concepts	& tools	℥ display-buffer	℥ Hooks		℥✖ - ELisp Topics		℥✖ - ELisp Types		℥ Elisp Build Tools	℥ ERT (regr-testing)	
Parsing tools, Indentation	℥ Xref Tools:	🔑 Indentation Styles	🔑 Language Servers		🔑 Tree-sitter		🔑 Xref-Backend		🔑 Xref-Frontend	🔑 Xref-Support	
Build Tools		℥ CMake 🔑🔑	℥ Make gmake		℥ Meson		℥ Ninja		℥ Nix	℥ Tup	
Data Serialization & Configuration		📄 CWL	📄 HCL/Terraform🔑🔑		📄 JSON 🔑🔑		📄 PKL 🔑🔑		📄 XML 🔑🔑	xmake	
Modelling		📄 ASN.1 asn1-mode	📄 MIB snmp-mode		📄 YANG		📄 YAML		📄 ZIGGY		
Other File Formats		Binary, Object, Executable Files			Log Files		RFC (RFC @ Wikipedia)			SSH files 🗑️ssh	
		℥ Changelog Files	Config/ini/toml... Files					RPM Files 🐧 (spec file format)		℥ X.509 Certificates	
Hardware Description Languages		🔑 Verilog 🔑🔑	🔑 VHDL 🔑🔑		🔑 Language Server & Tools for HDL 🔑🔑						
Lightweight Markup Languages		℥ AsciiDoc	℥ Markdown		℥ Org-Mode		℥ reStructuredText			℥ HamI 🔑🔑 future	
Graphics Markup		℥ Graphviz Dot	℥ MscGen		℥ PlantUML						
Programming Languages Major Modes		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	
Main Paradigm of Programming Languages - Actor Model: 📄 Array ℥ - Concatenative 📄 Concurrent: 📄 - Domain Specific 📄 - Dynamic d Extensible 📄 - Functional: 📄 Pure: 📄 - Generic 📄 homoiconic ℥ - Imperative: 📄 or no token - Object Oriented 📄 Procedural 📄 - Has Syntactic Macros: ℥ - Multi-paradigm 📄 Reflective - System Level 📄 - Theorem Proving (T) - 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.		℥ Ada 🔑🔑 📄📄	℥ D 📄📄📄		℥ Gambit 📄📄		℥ Janet 📄📄📄		℥ Pascal	Scala 🔑🔑	
		Agda 🔑🔑. (T)	℥ Dart 📄📄📄		℥ Gerbil 📄📄📄		℥ Java 🔑🔑		℥ Perl (perl5)	℥ Scheme 📄📄	
		℥ Algol	℥ Eiffel 🔑🔑 📄📄		℥ GNU Guile 📄📄		℥ Javascript 🔑🔑		PHP 🔑🔑 future	℥ Shen 📄📄 (T)	
		℥ AppleScript	Elm 🔑🔑 📄		℥ Gleam		℥ Julia 📄		℥ Pike d 📄📄📄	℥ Seed7 📄📄📄 📄	
		APL 🔑🔑	℥ Elixir 📄📄📄📄		℥ Go 📄		Kotlin 🔑🔑		Prolog 🔑🔑	SQL 🔑🔑	
		℥ Arc 📄📄	℥ Emacs Lisp		Groovy 🔑🔑		℥ LFE 📄📄📄📄		Pony 🔑🔑	℥ Smalltalk 🔑🔑 📄	
		℥ awk 📄	℥ Erlang 📄📄📄		℥ Haskell 📄		℥ Lua 📄📄📄		Purescript 🔑🔑📄📄	℥ Swift	
		℥ C 📄	℥ Factor 📄📄📄📄		Haxe 🔑🔑		Luerl 🔑🔑		℥ Python d📄📄📄📄	℥ Tcl 📄📄	
		C# 🔑🔑 future	FAUST 🔑🔑 future		℥ Hy (python) ℥		℥ M4		R 🔑🔑 📄📄📄📄	℥ Typescript 🔑🔑	
		℥ C++ 📄📄	Fennel 🔑🔑 future				℥ Modula		℥ Racket 📄📄	℥ UNIX Shell	
		℥ C3 📄	℥ Forth 📄				Mojo 🔑🔑 future		ReasonML 🔑🔑	℥ V	
		Carbon 🔑🔑 future	℥ Fortran				℥ NetRexx		℥ Rebol ℥	Vala 🔑🔑 future	
		℥ Chez 📄📄					℥ Nim 📄📄		Red 🔑🔑	℥ Zig 📄	
		℥ Chibi 📄📄					℥ Objective-C 🔑🔑		℥ REXX		
		℥ Chicken 📄📄					℥ OCaml 📄📄		Rocq 🔑🔑 future		
		℥ Clojure 📄📄					℥ Odin 📄		℥ Ruby		
		℥ Common Lisp ℥							℥ Rust 📄		
		Crystal 🔑🔑									
Future support for APL, Carbon, Crystal, Elm, Groovy, Haxe, Kotlin, Luerl, Pony, Prolog, Purescript, ReasonML, Rebol, Red, Scala, Typescript (based on my need for them or requests).											