PEL Topics Index

			i EE TOPK	JO IIIGOX			
Last updated on: 2025-05-07 Note: with PEL, type < f1> to open this PDF index.							
Emacs Reference Cards		These are links to the PDF version of official Eng PEL documents Emacs key bindings as well, the		glish version of the quick reference cards for GN			
With PEL, access these cards from Emacs with the <f11> ? e r key sequence.</f11>		Emacs Calc		Gnus Magit Cheatsheet		Org	Viper
See \mathbb{Z} Help/Info for more info.		Emacs survival card		Gnus booklet	Magit Ref-card	<u>Old</u>	<u>VIP</u>
➤ PEL Overview PEL license		This table holds links to the PEL file tables (hosted on Github as raw PDF files).					
PEL repo PEL Readme PEL Manual 2025-05-07		 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 pla	ce, you can browse throu	ugh all the PDFs and rea	ach a vast amount of info	ormation quickly.	
• PEL NEWS • Discussions	Emacs Mailing Lists	•				e. More help topics with <	2f11> ? p keys.
Terminal Multiplexers:	General Info ≻	✓ The symbols, colour coding and various other conventions are described in the ➤ Legend PDF. ➤ Legend ➤ Recommended Emacs User Option ➤ Themes Migrate from CRiSP					
GNU screen , Tmux	Startup >	<u> </u>			_		
Command Line Scripting Languages: bash, sh, zsh Cmdline: GNU readline, ls -l	PEL Code >	How to do it with PEL	_		iMenu/Speedbar su PEL Environment V		■PEL utilities
		€ macOS Fct Keys	PEL Naming Conve	ź			
OS Desktop Key Bindings (Bindings that don't clash with PEL)		W IIIdoo I ot Itaja	terminal settings	t towning a settings		16.04 Desk	top Keys
		0		Thouse Linux o Desktop Reys			
Feature Comparison		Completion Modes				§ Shells/Terminals Co	
Key Prefixes & Suffixes	S	∑ Modifier Keys	<u></u> ∑ ■Numkeypad	Keys - Fn	Keys - F11	Keys - F12	<u>≻PEL</u>
Emacs Features						s. The green links are mo	
 A <u>Guided Tour of Emacs</u>. <u>Awesome-Emacs</u> 		∑ Abbreviations	∑ Diff & Merge	<u>∑ Grep</u>	Marking	∑ Scrolling	<u> </u>
MELPA and GNU ELPA The tables listed at right described.	ihe Emacs	<u>» Align</u>	<u>∑ Dired</u>	∑ Help/Info	<u>∑ Menus</u>	∑ Search/Replace	T Templates
commands & key bindings for commands a key bindings for commands.	concepts &	∑ Auto-Completion	∑ Display - Lines	∑ Hide/Show	∑ Mode Line	∑ Sessions	∑ Text Modes
features. The cell is light-blue for light-red for minor mode specific	fic concepts.	∑ Autosave/Backup	∑ Drawing	<u>∑ Highlight</u> (colors)	<u>∑ Mouse</u>	∑ start Shells/REPLs	∑ Time Tracking
Grey cells are links into other pa important concepts.		<u> ∑ Bookmarks</u>	∑ Enriched Text	∑ ibuffer-mode	∑ Narrowing	∑ shell-mode	∑ Tramp ि
Emacs commands can be exect bound to key sequences. They		∑ Buffers	∑ Execute Cmds	∑ Indentation	∑ Navigation	∑ term-mode	<u>∑ Transpose</u> text
commands, their arguments and		∑ Case Conversions	∑ Exec Shell Cmds	∑ Input Method	∑ Object Files	∑ eat-mode	<u>∑X Treemacs</u>
sequences bound to them.Emacs Keys		∑ Close/Suspend	∑ Faces/Fonts	∑ Inserting Text	∑ Outline	∑ vterm-mode	∑ Undo/Redo/Repeat ▼ VCS Git *Magit
Numeric Arguments You can also:		∑ Comments ∑ Compilation Mode	∑P Fast Startup ∑ File Encoding	∑ Key-Chords ∑ Keyboard Macros	∑ Packages ∑ Projectile	∑X Smartparens ∑ Sorting	∑ VCS-Git XMagit ∑ VCS-Mercurial
Run Command by Name		∑ Compliation Mode ∑ Completion/Input	∑ File Encoding ∑ File-mngt	®IX- Lispy	<u> </u>	∑ Speedbar	∑ VCS-Mercurial ∑ VCS-Subversion
Emacs uses a concept of mode • Emacs Major and Minor Mo		∑ Completion/input ∑ Counting	∑ File-mngt ∑ File/Dir Variables	alta- Fisha		Speedbar Spell Checking	∑ Web
Major Modes	des	<u>∞M CUA</u>	∑ Fill/Justify		<u> </u>	SyntaxCheck SyntaxCheck	∑ Whitespace
Minor Modes Choosing Modes PEL provides several key sequences to toggle minor modes.		∑ Cursor	∑ Frames			<u> Z Oymaxoncox</u>	∑ Windows
		∑ Customize	<u>E i i unio</u>				∑ Xref - Cross Refs
		∑ Cut & Paste					<u> </u>
		≴ display-buffer	≴ - ELisp Types	<u>★ ERT</u> (regr-testing)	≴ Hooks		
XRef - Cross Reference			v	, , ,	, , , , , , , , , , , , , , , , , , ,	chanisms take advantag	e of various external
also: ∑ Xref	e ioois see					section. Also describes in	
		₫ Xref-Support	Xref-Frontend	Xref-Backend			Indentation Styles
Build Tools & Preproce	essor		<u>рї - М4</u>	भूर - Make gmake	<u>Bι̃ - Nix</u>	<u> </u>	
Data Serialization & Mo	odelling	© CWL	<u>©</u> <u>YAML</u>		S ASN.1 asn1-mode	S MIB snmp-mode	<u>S</u> <u>YANG</u>
Other File Formats		Changelog Files	Config/ini/toml Files	RFC (RFC @ Wikipedia)	RPM Files 40		M X.509 Certificates
Hardware Description La	<u>anguages</u>	Verilog 🚧	VHDL 🚧		(spec file format)		
Lightweight Markup Lan	<u>iguages</u>	M AsciiDoc	<u>Markdown</u>	M Org-Mode	<u>M reStructuredText</u>		
Graphics Markup		M Graphviz Dot	<u>M MscGen</u>	<u>M PlantUML</u>			
Programming Languages		Emacs has major mode	support for several pro	gramming languages. P	PEL extends Emacs supp	port for some of them (otl	hers are marked ##).
Main Paradigm of Programmi • Actor Model: Concar	ning Languages atenative ®	BEAM Programming	<u>Functional</u>	Javascript target	Pascal-style syntax	Lisp-like Languages	Stack Based
• Concurrent: © Domain Specific d • Dynamic & Extensible ©		Curly Bracket	Java Virtual Machine	ML Family	Lisp Family	Scheme Dialects	OS App Control
• Functional: ① Pure: ②		Ada 👑	BI-D (FA	PI - Gambit fm	₿፲ - Janet ①①®	BI - OCaml (i)f)	₽I - Rust S
Generic ① Imperative: ① or no token Object Oriented ② Procedural ② Has Syntactic Macros: ⑪ Multi-paradigm 》 Reflective System Level ③		Musta AppleScript	Dart ##		Java ##		-
							* *
		<u>Pl-Arc</u> (fm	Eiffel	PL - GNU Guile 🗇	PI - Javascript ##	<u>β</u> Ľ-Pascal	PI - Scheme (†m)
		<u>βί - awk</u>	PI - Elm ## F	PI - Gleam	PI - Julia	BI - Perl (perl5)	क्रॉ-Seed7 ﷺ © இ त
 The programming languages PEL are listed here in alphabe 	oetical order.	<u>\$\$ℓ - C</u>	<u>βί - Elixir</u> cmfA	<u> 1 μ - Go</u> Θ	Kotlin ##	<u>%I - Pike</u>	<u>Pii-Smalltalk</u> ₩ ⊚
 Emacs (and PEL) also provides basic support for some of the one PEL does not support and for other programming languages not listed here. 		<u> \$1 - C++</u>	⊈βl - Emacs Lisp	Groovy 🚧	<u>pι-lfe</u> ©mfA	Bi-Python dPOT	<u>βℓ-Swift</u>
		Carbon 🧱 🔇	<u>at - Erlang</u> ©fA	β Ι - Haskell 🕞	<u>βι -Lua</u> f @ P	भृर - Purescript 🚧 🕞	<u>\$1 - Tcl</u> f ①
	- Katha Luo	Bi - Chez fm	<u>β</u> <u>ι - Factor</u> (k) f 0 m	Haxe 🚧	β ί-Modula	<u>apt - Racket</u> ∱m	អ្ - Typescript 🚧
Future support for Crystal, Elm, Kotlin, Lua, Purescript, ReasonML, Typescript and documentation of support for Ada, Fortran, Javascript, Java, Modula, (based on my need for them or requests).		<u> PI - Chibi</u>	<u>aβι - Forth</u> (€	<u>ൂi - Hy</u> (python) ™	ֆῖ - NetRexx	ា្រ - ReasonML 🚧	ន្ទរ - UNIX Shell
		β ℓ - Chicken ∱®	Fortran 🚧		<u>ൂ≀ - Nim</u> @⊗	pι - REXX	<u> 191 - V</u>
		<u>βι - Clojure</u> ①			<u>al-Objective-C</u> ##	<u>aβt - Ruby</u>	<u>nt-Zig</u> ⊗
		Common Lisp fm					
		Crystal ##					