

# PEL Quick Access Topics Index

Last updated on: 2026-01-27

Note: 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.

Links to PDF version of official English version of the quick reference cards for [GNU Emacs](#) and popular external packages.

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

- [PEL](#) [Readme](#)  
 • [Repo](#) [License](#)  
 • [Manual](#) [NEWS](#)  
 • [Discussions](#)

- [Emacs Mailing Lists](#)  
[Contribute to Emacs](#)  
[EmacsConf](#)

This table holds links to all other [PEL topic oriented PDF table files](#) (hosted on Github).  
 ↗ For best user experience, use a browser like [Firefox](#) that can render PDF directly instead of downloading: all PDFs are heavily hyperlinked.  
 ↗ 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 >  
 Startup >  
 PEL Code >

<a href="#">&gt;Legend</a>	<a href="#">&gt;Recommended Emacs User Option</a>	<a href="#">&gt;Themes</a>	<a href="#">Migrate from CRISP</a>	
	<a href="#">Run Emacs daemon &amp; clients</a>	<a href="#">iMenu/Speedbar support</a>		
<a href="#">How to do it with PEL</a>	<a href="#">PEL Naming Conventions</a>	<a href="#">PEL Environment Variables</a>		<a href="#">PEL utilities</a>

## OS Desktop Key Bindings

(Bindings that don't clash with PEL)

## Feature Comparisons

### Key Prefixes & Suffixes

#### 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 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](#)
- [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.

[1979 EMACS Intro memo by R.M. Stallman](#)

### Emacs Lisp Ref concepts

#### & tools

<a href="#">display-buffer</a>	<a href="#">Hooks</a>	<a href="#">* - ELisp Topics</a>	<a href="#">* - ELisp Types</a>	<a href="#">Elisp Build Tools</a>	<a href="#">ERT</a> (regr-testing)
--------------------------------	-----------------------	----------------------------------	---------------------------------	-----------------------------------	------------------------------------

### Parsing tools, Indentation

#### Xref Tools:

<a href="#">Indentation Styles</a>	<a href="#">Language Servers</a>	<a href="#">Tree-sitter</a>	<a href="#">Xref-Backend</a>	<a href="#">Xref-Frontend</a>	<a href="#">Xref-Support</a>
------------------------------------	----------------------------------	-----------------------------	------------------------------	-------------------------------	------------------------------

## Build Tools

<a href="#">CMake</a>	<a href="#">Make</a>	<a href="#">gmake</a>	<a href="#">Meson</a>	<a href="#">Ninja</a>	<a href="#">Nix</a>	<a href="#">Tup</a>
-----------------------	----------------------	-----------------------	-----------------------	-----------------------	---------------------	---------------------

## Data Serialization & Configuration

<a href="#">CWL</a>	<a href="#">HCL/Terraform</a>	<a href="#">JSON</a>	<a href="#">PKL</a>	<a href="#">XML</a>	<a href="#">xmake</a>
---------------------	-------------------------------	----------------------	---------------------	---------------------	-----------------------

## Modelling

<a href="#">ASN.1 asn1-mode</a>	<a href="#">MIB snmp-mode</a>	<a href="#">YANG</a>		<a href="#">YAML</a>	
---------------------------------	-------------------------------	----------------------	--	----------------------	--

## Other File Formats

<a href="#">Binary, Object, Executable Files</a>	<a href="#">Log Files</a>	<a href="#">RFC</a> (RFC @ Wikipedia)		<a href="#">SSH files</a>	<a href="#">ssh</a>
--	---------------------------	---------------------------------------	--	---------------------------	---------------------

### Changelog Files

<a href="#">Config/ini/toml... Files</a>		<a href="#">Recursive Edit</a>	<a href="#">SyntaxCheck</a>	
--	--	--------------------------------	-----------------------------	--

<a href="#">RPM Files</a> (spec file format)		<a href="#">Registers</a>		<a href="#">Windows</a>	
--	--	---------------------------	--	-------------------------	--

				<a href="#">Writing Tools</a>	
--	--	--	--	-------------------------------	--

				<a href="#">Xref - Cross Refs</a>	
--	--	--	--	-----------------------------------	--

## Hardware Description Languages

<a href="#">Verilog</a>	<a href="#">VHDL</a>	<a href="#">Language Server &amp; Tools for HDL</a>			
-------------------------	----------------------	---	--	--	--

## Lightweight Markup Languages

<a href="#">AsciiDoc</a>	<a href="#">Markdown</a>	<a href="#">Org-Mode</a>	<a href="#">reStructuredText</a>		<a href="#">Haml</a>
--------------------------	--------------------------	--------------------------	----------------------------------	--	----------------------

## Graphics Markup

<a href="#">Graphviz Dot</a>	<a href="#">MscGen</a>	<a href="#">PlantUML</a>		
------------------------------	------------------------	--------------------------	--	--

## Programming Languages Major Modes

<a href="#">BEAM Programming</a>	<a href="#">Functional</a>	<a href="#">Javascript target</a>	<a href="#">Pascal-style syntax</a>	<a href="#">Lisp-like Languages</a>	<a href="#">Stack Based</a>
----------------------------------	----------------------------	-----------------------------------	-------------------------------------	-------------------------------------	-----------------------------

<a href="#">Curly Bracket</a>	<a href="#">Java Virtual Machine</a>	<a href="#">ML Family</a>	<a href="#">Lisp Family</a>	<a href="#">Scheme Dialects</a>	<a href="#">OS App Control</a>
-------------------------------	--------------------------------------	---------------------------	-----------------------------	---------------------------------	--------------------------------

## Main Paradigm of Programming Languages

<a href="#">Actor Model</a>	<a href="#">Array</a>	<a href="#">Concurrent</a>	<a href="#">Domain Specific</a>	<a href="#">Dynamic</a>	<a href="#">Extensible</a>
-----------------------------	-----------------------	----------------------------	---------------------------------	-------------------------	----------------------------

<a href="#">Functional</a>	<a href="#">Pure</a>	<a href="#">Generic</a>	<a href="#">Imperative</a>	<a href="#">or no token</a>	<a href="#">Object Oriented</a>
----------------------------	----------------------	-------------------------	----------------------------	-----------------------------	---------------------------------

<a href="#">Procedural</a>	<a href="#">Has Syntactic Macros</a>	<a href="#">Multi-paradigm</a>	<a href="#">Reflective</a>	<a href="#">System Level</a>	<a href="#">Paradigm</a>
----------------------------	--------------------------------------	--------------------------------	----------------------------	------------------------------	--------------------------

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, Elm, Groovy, Haxe, Kotlin, Pony, Purescript, ReasonML, Rebol, Red, Scala, Typescript (based on my need for them or requests).					
--	--	--	--	--	--

--	--	--	--	--	--

--	--	--	--	--	--

--	--	--	--	--	--

--	--	--	--	--	--