↖ Topic Index

Emacs Support for Scheme 🚧

Description	Keystroke	Function	Note
Scheme Programming Language Support	🚧 This page is not written yet, except for some basic information. More information will be provided later. PEL activates Scheme support when the **pel-use-scheme** user-option is turned on (**t**). PEL provide extra support for the Scheme programming language and its various implementations by providing access to the following external packages: 📦 The **geyser** external package. PEL activates it when the **pel-use-geiser** user-option is turned on (**t**). 📦 The **quack** external package. PEL activates it when the **pel-use-quack** user-option is turned on (**t**). Emacs built-in Scheme support recognizes the Scheme dialect like Gambit Scheme, Gerbil Scheme and Racket. When pel-use-scheme is turned on PEL provides the **<f12>** keys for the Scheme buffers for this basic support. If you need more specialized support for the Theme dialect of your choice, PEL also provides some. See the relevant PDF for those, including: • **𝔓𝔏 - Gambit** • **𝔓𝔏 - Gerbil** • **𝔓𝔏 - GNU Guile** • **𝔓𝔏 - Racket** • PEL provides Lispy support for Scheme when the scheme-mode is added to the list specified by **pel-modes-activating-lispy** user-option. See **𝔓𝔏- Lispy**		
Open this PDF file. See also: **∑ Help/Info**	**<f11> SPC C-s C-s <f1>** **<f12> <f1>**	(**pel-help-pdf** &optional OPEN-WEB-PAGE)	Open the **𝔓𝔏 - Scheme** local PDF. If the prefix argument (like **C-u** or **M--**) is used, then it opens the remote GitHub hosted raw PDF instead. If the **pel-flip-help-pdf-arg** user-option is set it's the other way around.
∑ Customize PEL Scheme support	**<f11> SPC C-s C-s <f2>** **<f12> <f2>**	(**pel-customize-pel** &optional OTHER-WINDOW)	Customize PEL Scheme support. • If OTHER-WINDOW is non-nil (use **C-u**), display in another window.
∑ Customize Emacs Scheme support	**<f11> SPC C-s C-s <f3>** **<f12> <f3>**	(**pel-customize-library** &optional OTHER-WINDOW)	Customize Emacs Scheme support: scheme, geiser, quack. • If OTHER-WINDOW is non-nil (use **C-u**), display in another window.