## Emacs Support for Clojure ##

Description	<u>Keystroke</u>	Function	Note
Clojure Programming Language Support	This page is not written yet, except for some basic information. More information will be provided soon.  PEL activates Clojure support when the pel-use-clojure user-option is turned on (t).  PEL provide extra support for the Clojure programming language and its various implementations by providing access to the following external packages:  The cider external package. PEL activates it when the pel-use-cider user-option is turned on (t).  The clj-refactor.el external package. PEL activates it when the pel-use-clj-refactor user-option is turned on (t).  The clojure-snippets external package. PEL activates it when the pel-use-clojure-snippets user-option is turned on (t).  PEL provides Lispy support for Clojure when the clojure-mode is added to the list specified by pel-modes-activating-lispy user-option. See PI- Lispy		
Open this PDF file. See also: <u>∑ Help/</u> Info	<f11> SPC C-j <f1></f1></f11>	WEB-PAGE)	Open the <u>Nt - Clojure</u> local PDF. If the prefix argument (like <b>C-u</b> or <b>M</b> ) is used, then it opens the remote GitHub hosted raw PDF instead. If the <b>pel-flip-help-pdf-arg</b> user-option is set it's the other way around.
	<f12> <f1></f1></f12>		
∑ Customize PEL Clojure support	<f11> SPC C-j <f2></f2></f11>		Customize PEL Clojure support.  • If OTHER-WINDOW is non-nil (use <b>C-u</b> ), display in another window.
	<f12> <f2></f2></f12>		
∑ Customize   Emacs Clojure   support	<f11> SPC C-j <f3></f3></f11>		Customize Emacs Clojure support: clojure, cider, cljr.  • If OTHER-WINDOW is non-nil (use <b>C</b> - <b>u</b> ), display in another window.
	<f12> <f3></f3></f12>		