









Emacs Support for Clojure

Description	Keystroke	Function	Note
Clojure Programming Language Support	<div>  This page is not written yet, except for some basic information. More information will be provided soon. </div> <div>  PEL activates Clojure support when the pel-use-clojure user-option is turned on (t). </div> <div> PEL provide extra support for the Clojure programming language and its various implementations by providing access to the following external packages: </div> <div>  The cider external package.  PEL activates it when the pel-use-cider user-option is turned on (t). </div> <div>  The clj-refactor.el external package.  PEL activates it when the pel-use-clj-refactor user-option is turned on (t). </div> <div>  The clojure-snippets external package.  PEL activates it when the pel-use-clojure-snippets user-option is turned on (t). </div> <div> • PEL provides Lispy support for Clojure when the clojure-mode is added to the list specified by pel-modes-activating-lispy user-option. See ℥L - Lispy </div>		
Open this PDF file. See also: ℥ Help/Info	<div><f11> SPC C-j <f1></div> <div><f12> <f1></div>	<div>(pel-help-pdf &optional OPEN-WEB-PAGE)</div>	Open the ℥L - Clojure PDF using method specified by the pel-open-pdf-method user-option or the alternate one if a command prefix (like C-u) was used.
℥ Customize PEL Clojure support	<div><f11> SPC C-j <f2></div> <div><f12> <f2></div>	<div>(pel-customize-pel &optional OTHER-WINDOW)</div>	Customize PEL Clojure support. <ul style="list-style-type: none"> If OTHER-WINDOW is non-nil (use C-u), display in another window.
℥ Customize Emacs Clojure support	<div><f11> SPC C-j <f3></div> <div><f12> <f3></div>	<div>(pel-customize-library &optional OTHER-WINDOW)</div>	Customize Emacs Clojure support: clojure, cider, cljr. <ul style="list-style-type: none"> If OTHER-WINDOW is non-nil (use C-u), display in another window.