

Emacs support for Nim

Description	Keystroke	Function	Note
Nim Editing	Emacs does not provide any built-in mode for Nim. 📦 The nim-mode external package is required. 📦 PEL installs and activates it when the pel-use-mode user-options is turned on (set to t). <ul style="list-style-type: none">As of Emacs 30.2 there is no Tree-Sitter aware mode for Nim implemented yet. See ⓘ Tree Sitter and 📦 Tree-sitterWhen pel-use-nim is turned on the <f11> SPC n prefix is made available. 🚧 PEL support for Nim is preliminary. More commands should be provided and documented.		
Last updated on:	2025-10-15		
Open this PDF file. See also: ⓘ Help/Info	<f11> SPC n <f1> <f12> <f1>	(pel-help-pdf &optional OPEN-WEB-PAGE)	Open the ℹ - Nim 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 nim support	<f11> SPC n <f2> <f12> <f2>	(pel-customize-pel &optional OTHER-WINDOW)	Customize PEL nim support. <ul style="list-style-type: none">If OTHER-WINDOW is non-nil (use C-u), display in another window.
ⓘ Customize Emacs nim support	<f11> SPC n <f3> <f12> <f3>	(pel-customize-library &optional OTHER-WINDOW)	Customize Emacs nim support: nim <ul style="list-style-type: none">If OTHER-WINDOW is non-nil (use C-u), display in another window.
Select nim-mode for extension-less file 👉 The <f12> key is available only until a PEL controlled major mode is activated. Then it becomes a buffer prefix key.	<f12>	(pel-as &optional FORCE)	Inside a fundamental-mode buffer, interactively select major mode for the buffer. Re-do it with arg. 👉 see Create extension-less executable scripts with PEL . This command is mostly used to set the major mode of a buffer in fundamental-mode', when the <f12> key binding is available for it. After being used once in a buffer the major mode is selected and the PEL key binding will not be available when PEL supports the major mode. For Nim file, select nim . It will insert a shebang line specified by 📦 pel-nim-shebang-line user option. PEL defines the (as &optional FORCE) alias unless 📦 pel-has-alias-as user-option is set to nil. You can use M-x as to invoke it.
Show PEL setup for Nim	<f12> ? <f11> SPC n ?	(pel-nim-setup-info &optional APPEND)	Display Sim setup information inside a "pel-nim-info" buffer with buttons providing quick access to the customization buffer of each variable shown. The information shown includes the value and interpretation of: <ul style="list-style-type: none">the user options controlling indentation and hard tab width rendering. To append information in the buffer instead of clearing the previous content type any prefix argument (such as C-u) before the command keystroke.
Generic code skeletons <ul style="list-style-type: none">tempo skeletons See also: <ul style="list-style-type: none">ⓘ Inserting TextT Templates	Several mechanisms have been developed to allow easy insertion of predefined text in Emacs. ⚠ PEL does not yet define skeletons for Nim. You can use the generic one. <ul style="list-style-type: none">Emacs provides the built-in skeleton mechanism and the tempo skeletons.<ul style="list-style-type: none">PEL supports both. They are used a little bit differently. PEL provides generic tempo skeletons you can use for Nim until PEL adds Nim-specific skeletons.PEL provides key bindings to the tempo skeletons: the generic code templates, accessible via the <f6> prefix key, and the language-specific code templates, accessible via the <f12> key prefix.		
ⓘ Customize PEL Text Insertions control for Nim code skeletons.	<f6> <f2> <f12> <f12> <f2>	(pel-customize-pel &optional OTHER-WINDOW) (pel-customize-generic-skels &optional OTHER-WINDOW)	Open the customization groups that control the format of the various skeletons including the generic skeleton used by the <f6> h key and the <f12><f12> h key (see below). <ul style="list-style-type: none">If OTHER-WINDOW is non-nil (use C-u) , display in other window.
Insert generic file module header block — Language agnostic After inserting the template, navigate though areas that must be filled with: <ul style="list-style-type: none">tempo-forward-mark: C-c .tempo-backward-mark: C-c ,	<f6> h <f12> <f12> h	(pel-generic-file-header)	Insert a file header block at the top of the file. Works only for buffer visiting a file. ⚠ The command key binding <f6> h is available only 1 second after Emacs has started. ⚠ As mentioned above PEL does not yet define Nim-specific skeletons, this uses the generic one.
	👉 Specify the format of the header via the user-options in the pel-pkg-generic-code-style customization group accessible via <f6> <f2> <ul style="list-style-type: none">Inside a Nim buffer, <f12> <f2> provides access to the following customization groups: 👉 After inserting a template, use tempo-forward-mark and tempo-backward-mark to move to the beginning of each section that must be filled.		
Toggle pel-tempo-mode	<f6> SPC <f12> <f12> SPC	(pel-tempo-mode &optional ARG)	Toggle PEL tempo mode on/off. PEL tempo mode activates C-c . and C-c , , as well as to C-c C-- and C-c C- , key bindings to navigate across tempo mark hot-spots. When pel-tempo-mode is active the pel-tempo-mode lighter (🔦) is shown on the status bar. The second set of keys are only available in graphics mode. 👉 The pel-generic-file-header command inserts the text using a tempo skeleton: the PEL tempo mode is automatically activated by typing <f6> h .
Expand any tag in template Note: PEL default skeleton does not use tags.	<f6> <f12> <f12> <f12> <f12>	(tempo-complete-tag &optional SILENT)	Look for a tag and expand it. All the tags in the tag lists in ' tempo-local-tags ' (this includes 'tempo-tags') are searched for a match for the text before the point. The way the string to match for is determined can be altered with the variable 'tempo-match-finder'. If 'tempo-match-finder' returns nil, then the results are the same as no match at all. <ul style="list-style-type: none">If a single match is found, the corresponding template is expanded in place of the matching string.If a partial completion or no match at all is found, and SILENT is non-nil, the function will give a signal.If a partial completion is found and 'tempo-show-completion-buffer' is non-nil, a buffer containing possible completions is displayed.
Comments			
Toggle display of comments in buffer or active region See also: ⓘ Comments	<f11> ; ;	(hide/show-comments-toggle &optional START END)	Toggle hiding/showing of comments in the active region or whole buffer. <ul style="list-style-type: none">If the region is active then toggle in the region. Otherwise, in the whole buffer. 📦 This requires the hide-comnt.el package (see ⓘ Comments). 📦 PEL activates it when the pel-use-hide-comnt user option is t .

Emacs & Nim — References

Document	Notes
Information about Nim	<ul style="list-style-type: none"> Nim @ Wikipedia Nim home