

Inserting Text

Description	Keystroke	Function	Note
<div>Inserting Text</div> <div>See Σ Smartparens</div>	<div>The commands described in this table insert specialized text at point (cursor) location. This includes:</div> <div><ul style="list-style-type: none">Customization to control automatic insertion of time stamp, update of copyright notice.</div> <div>1. <i>Simple command based text insertions:</i><ul style="list-style-type: none">PEL specialized commands to insert formatted text like time stamps, file path , file name, copyright notice, Greek letters, commented lines, etc... The lice external package activated by pel-use-lice user option, used to insert open source licence text. The smart-dash external package activated by pel-use-smart-dash, used to automatically convert dash into underscore when typing. The smartparens external package activated by pel-use-smartparens to provide automatic insertion of balanced block pairs for code.</div> <div>2. <i>Specialized template-based text insertion:</i><ul style="list-style-type: none">PEL tempo skeletons based templates for generic & specialized boilerplate file sections: file, class, function header, document section header. The yasnippet external package activated by pel-use-yasnippet to insert code from predefined snippets. The yasnippet-snippets external package activated by pel-use-yasnippet-snippets which provides a large amount of snippets.</div> <div>3. <i>Hydra-based insertion of Greek letters:</i><ul style="list-style-type: none"> The hydra external package. activated by pel-use-hydra and pel-activate-hydra-for-greek.</div>		
<div>Open this PDF file.</div> <div>See also: Σ Help/Info</div>	<div><ul style="list-style-type: none"><code><f11> i <f1></code><code><f11> y <f1></code><code><f11> _ <f1></code></div>	<div>(pel-help-pdf &optional OPEN-WEB-PAGE)</div>	<div>Open the Σ Inserting Text 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.</div>
<div>Σ Customize PEL Text Insertions control</div>	<code><f11> i <f2></code>	<div>(pel-customize-pel &optional OTHER-WINDOW)</div>	<div>Customize PEL text insertion support: lice, smart-dash, smartparens, tempo, time-stamp, yasnippet. Also pel-activate-f9-for-greek (see below).</div> <div><ul style="list-style-type: none">If OTHER-WINDOW is non-nil (use C-u), display in other window.</div>
<div>Σ Customize Emacs Text Insertions control</div>	<code><f11> i <f3></code>	<div>(pel-customize-library &optional OTHER-WINDOW)</div>	<div>Customize Emacs text insertion support: lice, smart-dash, tempo, time-stamp, yasnippet</div>
<div>Insert Greek Letter</div> <div>See also: Σ Input Method</div>	<div><ul style="list-style-type: none"><code><f9></code> <code><f6> g</code> </div>	<div>Insert a greek letter: type <code><f9></code> followed by a key in [a-zA-Z] range inserts the Unicode character for the equivalent Greek letter.</div> <div>Examples: <code><f9> a</code> inserts α <code><f9> b</code> inserts β <code><f9> A</code> inserts Α <code><f9> B</code> inserts Β <code><f9> l</code> inserts λ</div> <div><ul style="list-style-type: none">The insertions work everywhere insert is allowed, including in response to prompts.Use <code><f9> C-h</code> or which-key mode and type <code><f9></code> to see all keys.</div> <div> The <code><f9></code> key binding is only available when the pel-activate-f9-for-greek user-option is turned on.</div> <div>The <code><f6> g</code> binding is always available.</div> <div> This is not a command bound to a key: it's an additional set of bindings added to Emacs key-translation-map.</div>	
<div>Start pel-Σgreek Hydra</div> <div><ul style="list-style-type: none">Quickly type succession of Greek characters</div>	<code><f7> <f6> <f6></code>	<div>Start the Greek letter Hydra.</div> <div><ul style="list-style-type: none">After typing <code><f7> <f6> <f6></code> type one of the Meta letter keys in the hydra to insert a Greek character. Type any other character to insert them, latin letters, digits, punctuation characters, etc...In terminal mode the cursor keys may not work because they are often encoded using Esc keys with is mapped to Meta.</div> <div> Requires the hydra external package. activated by pel-use-hydra.</div> <div> You must also set pel-activate-hydra-for-greek to t to activate this hydra.</div> <div><ul style="list-style-type: none">Exit the hydra by typing <code><f7></code></div> <div><pre>File Edit Options Buffers Tools Defs Lisp-Interaction Help With this Hydra, you can type Greek text by pressing the Meta keys. Με αυτό το Hydra, μπορείτε να πληκτρολογήσετε ελληνικό κείμενο πατώντας τα πλήκτρα Meta. No prefix is active: all digit keys can be typed: 0123456789 ;; M-u for undo is not available: use ``<f11> u u`` instead. ;; In terminal mode the cursor keys may not work though. Use C-b, C-f, C-n, C-p instead. Exit the Hydra by typing <f7> -UUU:**--F1 *scratch* All (9,29) (Lisp Interaction ⚡ WK Anzu Fly ² ElDoc) 12:56pm ----- [M-a]: α, [M-b]: β, [M-c]: γ, [M-d]: δ, [M-e]: ε, [M-f]: φ, [M-g]: υ, [M-h]: η, [M-i]: ι, [M-j]: φ, [M-k]: κ, [M-l]: λ, [M-m]: μ, [M-n]: ν, [M-o]: ο, [M-p]: π, [M-q]: θ, [M-r]: ρ, [M-s]: σ, [M-t]: τ, [M-u]: υ, [M-w]: ω, [M-x]: ξ, [M-y]: ψ, [M-z]: ζ, [M-A]: Α, [M-B]: Β, [M-C]: Χ, [M-D]: Δ, [M-E]: Ε, [M-F M-J]: Φ, [M-G]: Γ, [M-H]: Η, [M-I]: Ι, [M-K]: Κ, [M-L]: Λ, [M-M]: Μ, [M-N]: Ν, [M-O]: Ο, [M-P]: Π, [M-Q]: Θ, [M-R]: Ρ, [M-S]: Σ, [M-T]: Τ, [M-U]: Υ, [M-W]: Ω, [M-X]: Ξ, [M-Y]: Ψ, [M-Z]: Ζ, [<f7>]: cancel.</pre></div>	
<div>Insert file/directory name</div>	<div>The following commands insert the name of the file or directory of the current file or file in specified window.</div>		
<div>Insert, at point, name of:</div> <div><ul style="list-style-type: none">current filename by defaultname of file in window identified by command numeric argument</div>	<div><ul style="list-style-type: none"><code><f11> i f</code><code><f6> f</code></div>	<div>(pel-insert-filename &optional N USE-TILDE DIR-ONLY)</div>	<div>Insert the file name of the currently edited file at point.</div> <div><ul style="list-style-type: none">By default, or with 1, insert filename of current buffer with complete absolute path.With a numeric argument: identify the window where the file name is taken:<ul style="list-style-type: none">8: up, 2: down, 4: left, 6: right. Any other number identifies the current window.When the numeric argument is positive the file with complete absolute path is inserted,With negative numeric argument the path is omitted.</div>
<div><ul style="list-style-type: none">If file is in user home, use ~ at the beginning</div>	<div><ul style="list-style-type: none"><code><f11> i M-f</code><code><f6> M-f</code></div>	<div>(pel-insert-filename-wtilde &optional N)</div>	<div>Same as the above command, except that if the file is located in the current user home, insert the Unix-style tilde character ~ in place of the user home directory name.</div>
<div>Insert, at point, name of:</div> <div><ul style="list-style-type: none">current dirname by defaultdirname of file in window identified by command numeric argument</div>	<div><ul style="list-style-type: none"><code><f11> i C-f</code><code><f6> C-f</code></div>	<div>(pel-insert-dirname &optional N USE-TILDE)</div>	<div>Insert the directory path name of the currently edited file at point.</div> <div><ul style="list-style-type: none">By default, or with 1, insert directory name of file in current buffer.With a numeric argument: identify the window where the directory name is taken:<ul style="list-style-type: none">8: up, 2: down, 4: left, 6: right. Any other number identifies the current window.When the numeric argument is positive the file with complete absolute path is inserted.</div>
<div><ul style="list-style-type: none">If file is in user home, use ~ at the beginning</div>	<div><ul style="list-style-type: none"><code><f11> i C-M-f</code><code><f6> C-M-f</code></div>	<div>(pel-insert-dirname-wtilde &optional N)</div>	<div>Same as the above command, except that if the file is located in the current user home, insert the Unix-style tilde character ~ in place of the user home directory name.</div>
<div>Insert time & date</div>	<div>The following commands insert time stamps of specific formats.</div>		
<div>Insert current date</div>	<div><ul style="list-style-type: none"><code><f11> i d</code><code><f6> d</code></div>	<div>(pel-insert-current-date &optional UTC)</div>	<div>Insert current date (only, no time) at point.</div> <div><ul style="list-style-type: none">Local by default, UTC if C-u prefix used.</div>
<div>Insert current date & time</div>	<div><ul style="list-style-type: none"><code><f11> i D</code><code><f6> D</code></div>	<div>(pel-insert-current-date-time &optional UTC)</div>	<div>Insert current date and time at point.</div> <div><ul style="list-style-type: none">Local by default, UTC if C-u prefix used.</div>
<div>Insert time stamp</div>	<div><ul style="list-style-type: none"><code><f11> i t</code><code><f6> t</code></div>	<div>(pel-insert-iso8601-timestamp &optional UTC)</div>	<div>Insert ISO 8601 conforming abbreviated YYYY-MM-DD hh:mm:ss format timestamp.</div> <div><ul style="list-style-type: none">Local by default, UTC if C-u prefix is used.</div>
<div>Insert software license text</div>	<div><ul style="list-style-type: none"><code><f11> i L</code><code><f6> L</code></div>	<div>(lice NAME)</div>	<div>Insert license and headers at point.</div> <div>Prompts for license NAME, which is a license template name like “mit”, “gpl-3.0”, etc... The list is available with TAB completion: hit TAB on prompt to get the complete list of templates.</div> <div> Requires the lice external package. PEL activates it if pel-use-lice user option is t.</div>

Description	Keystroke	Function	Note
<p>Automatic File Time Stamp on file save</p> <p>References:</p> <ul style="list-style-type: none"> TimeStamps @ EmacsWiki Change time stamp format in: <ul style="list-style-type: none"> markdown file reStructuredText file <p>See also: 🔗 File mngt</p>	<p>Emacs has a built-in automatic time-stamping of files. It must be activated by adding the time-stamp function to the before-save-hook variable. This can either be done via Emacs customization system or explicitly inside your init file with the following code:</p> <pre>(add-hook 'before-save-hook 'time-stamp)</pre> <ul style="list-style-type: none"> The time stamp will be added to files that contain, inside their first 8 lines, a line that looks like one of the following: <ul style="list-style-type: none"> Time-stamp: <> Time-stamp: " " <p>👉 You can, however change these defaults and get Emacs to update all sorts of time stamp formats, even inside source code statements:</p> <p>🔧 Emacs controls automatic insertion of timestamp with the following variables:</p> <ul style="list-style-type: none"> time-stamp-pattern consists of 4 parts, each one controlled by a variable: <ul style="list-style-type: none"> time-stamp-line-limit : identifies where in the file the time stamp can be located. Defaults to 8: the first 8 lines. time-stamp-start: identifies the text pattern that precedes the time stamp. time-stamp-end: identifies the end of the time stamp. time-stamp-format specifies the format of the time stamp. <ul style="list-style-type: none"> Something like "%:y-%02m-%02d %02H:%02M:%02S %u" to specify the date and time in ISO format, with the user login's name. time-stamp-time-zone specifies the time zone selection: <ul style="list-style-type: none"> nil : Emacs local time t : Universal time wall : system wall clock time TZ : controlled by a TZ environment variable <p>The time-stamp-format and time-stamp-time-zone variables can be set in your init file or via the Emacs customization system.</p> <ul style="list-style-type: none"> They are defined in the time-stamp customization group. 👉 To change the format or the pattern preceding or after the automatically updated time stamp, it is best to use file local variables: this will allow automatic time stamp updates in files with various formats. As an example, see the top and end of the PEL manual raw format file. <p>⚠️ By default, the time-stamp string must be placed within the first 8 lines of the file, otherwise it will not be updated automatically.</p> <ul style="list-style-type: none"> If you want it located somewhere else in your file set the time-stamp-line-limit file local variable. <p>🔧 PEL provides the extra user-option to control the automatic generation of time-stamps:</p> <ul style="list-style-type: none"> pel-update-time-stamp user-option controls whether time-stamps are automatically update time stamps in all files where a valid time-stamp corresponding to Emacs settings as described above. Set it to t (the default) to allow automatic time stamp updates. Set it to nil to prevent them. You can also toggle it globally for the current editing session by using the <f11> f M-t key sequence. <p>👉 To insert a non-updatable time stamp, the PEL package provides a set of text insert commands which include inserting a time stamp .</p>		
<p>Update file time stamp</p> <p>See also: 🔗 File mngt</p>	<f11> f t	(time-stamp)	<p>Force update the time stamp string(s) in the current buffer.</p> <ul style="list-style-type: none"> Updates a time stamp of format recognized by <i>Emacs current settings</i> even when automatic time-stamp update is off. More information about the “<i>Emacs current settings</i>” in the description block above.
<p>Toggle time stamp automatic update</p>	<f11> f M-t	(time-stamp-toggle-active &optional ARG)	<p>Toggle ‘time-stamp-active’, setting whether <f11> f t updates a buffer.</p> <ul style="list-style-type: none"> With ARG, turn time stamping on if and only if arg is positive.
<p>Inserting & Automatically Updating Copyrights</p>	<p>Emacs has built-in support for insertion and update of copyright notices inside files.</p> <ul style="list-style-type: none"> Two commands, shown below, are provided to manually insert or update the file's copyright notice. The copyright notice can be automatically updated by adding the copyright-update function to the list of before-save-hook variable with the following code: <pre>(add-hook 'before-save-hook 'copyright-update)</pre> <p>⚠️ To be automatically updated, the copyright notice must be placed within an area at the beginning of the file specified by the value of the copyright-limit variable, normally defined as the first 2000 characters. This variable is customizable.</p>		
<p>Insert copyright notice</p> <p>See also: 🔗 File mngt</p>	<f11> i c	(copyright &optional STR ARG)	<p>Insert a copyright by \$ORGANIZATION notice at cursor.</p> <ul style="list-style-type: none"> If the ORGANIZATION environment variable is not available, Emacs prompts for it.
<p>Update file's copyright notice</p>	<f11> i M-c	(copyright-update &optional ARG INTERACTIVEP)	<p>Update copyright notice to indicate the current year.</p> <ul style="list-style-type: none"> With prefix ARG, replace the years in the notice rather than adding the current year after them. <p>If necessary, and 'copyright-current-gpl-version' is set, any copying permissions following the copyright are updated as well.</p>
<p>⚠️ Only update exiting notice.</p> <ul style="list-style-type: none"> Does not create one if it's missing. 	<p>⚠️ copyright-update does not warn if there is no copyright in the current buffer to update. It does not create a missing notice.</p> <p>👉 📄 If you want automatic copyright notice updates when a modified buffer is saved, set the pel-update-copyright user option to t.</p> <ul style="list-style-type: none"> Without PEL add the following inside your init.el file: <pre>(add-hook 'before-save-hook 'copyright-update)</pre>		
<p>Insert Commented Lines</p>	<p>The following commands help insert commented lines or just underlines the current line of text using the character corresponding to one of the adornment level used for reStructuredText sections. The strings are commented according to the major mode of the current buffer. If the buffer has no identified comment strings, the command prompts for them the first time it is used in that type of buffer.</p> <p>The following commands are also listed in the 🔗 Comments table.</p>		
<p>Insert commented line</p> <p>See also: 🔗 Comments</p>	<ul style="list-style-type: none"> <f11> i 1 <f6> 1 	(pel-insert-line &optional LINELEN)	<p>Insert a (commented) line before/at current line.</p> <ul style="list-style-type: none"> If point is at the beginning of the line insert it there. If point is in the middle of a line, move point at beginning of line before inserting it. The number of dash characters of the line is specified by LINELEN: <ul style="list-style-type: none"> If LINELEN is not specified the buffer's fill-column value is used. It supports several programming and markup language and uses the comment style identified by the file extension. If the comment style is unknown the command prompts for one. <p>👉 fill-column is customizable and can be used as a file or directory variable.</p>
<p>Comment-underline current line with level adornment 1-9</p>	<f11> _ ☒	(pel-commented-adorn-1)	<p>Insert a commented level-x reST line adornment at point.</p> <ul style="list-style-type: none"> ☒ := 1 to 9 for levels 1 to 9
<p>Comment-underline current line with level 10 adornment</p>	<f11> _ 0	(pel-commented-adorn-10)	<p>Insert a commented level-10 reST line adornment at point.</p>
<p>To-Do Notes</p>			
<p>Insert To-Do note</p>	<ul style="list-style-type: none"> <f11> i T <f6> T 	(pel-insert-todo-note)	<p>Insert a to-do note template comment that contains its creation date and the author's name.</p> <ul style="list-style-type: none"> The note is inside a comment and uses a fixed format the is enclosed inside square brackets that identifies its beginning and end, the tag <i>:todo:</i> followed by a ISO-8601 compliant date and the authors's name. Point is left where the note must be written.

Inserting Text — References

Topic & link	Description
GNU Emacs Manual: Time Stamps	
Smart-Dash Mode homepage	A description of this extremely useful mode and why it was created.