# Text Modes

| Operation | Keystroke | Function | Note |
|---|---|---|---|
| **Text Modes** | | Emacs has several other text modes that deal with whitespaces.  See the **Text Whitespace** table for more information.<br><br>Note that superword-mode and subword-mode can be quite useful when editing source code where underscores are used to separate words in symbols or when CamelCase is being used. | |
| **Switch Insert/Overwrite mode** | • `<insert>`<br>• `<f11> t o`<br>• `` <f11> ` `` | (**overwrite-mode** &optional ARG) | Toggles the overwrite mode on/off<br>• The `<insert>` key is not available in macOS keyboards.<br>• With a prefix argument ARG, enable Overwrite mode if ARG is positive, and disable it otherwise. |
| **Binary file overwrite only mode** | `<f11> t O` | (**nhexl-overwrite-only-mode** &optional ARG) | Minor mode where text is only overwritten.<br>Insertion/deletion is avoided where possible and replaced by overwriting existing text, if needed with 'nhexl-overwrite-clear-byte'.<br>👉 The nhexl mode must first be activated to edit the file in binary mode.<br>📦 Requires the nhexl-mode package.<br>☑ PEL activates this when the **pel-use-nhexl** customize variable is set to **t**. |
| **Make Info control text visible/invisible (toggle visible mode)** | `<f11> t m v` | (**visible-mode** &optional ARG) | Toggle making all invisible text temporarily visible (Visible mode).<br>With a prefix argument ARG, enable Visible mode if ARG is positive, and disable it otherwise.  Useful for developing info files, where some characters are not visible by default.  Same in Org-mode (for example to show everything, or to show the syntax of links without expanding anything). |
| **Toggle subword-mode** | `<f11> t m b` | (**subword-mode** &optional ARG) | Toggle subword-mode: a minor mode that treats CamelCase as distinct words.<br>• With a prefix argument ARG, enable Subword mode if ARG is positive, and disable it otherwise. |
| **Toggle superword-mode** | `<f11> t m p` | (**superword-mode** &optional ARG) | Toggle superword-mode: a minor mode that treats snake_case as one word.  In Lisp, '-' and '_' are treated part of words.<br>• With a prefix argument ARG, enable Superword mode if ARG is positive, and disable it otherwise. |
| **Toggle sentence separators between 1 or 2 spaces** | `<f11> t m s` | (**pel-toggle-sentence-end**) | Toggle definition of end of sentence between 2 and 1 space character (to help text filling).  This has an impact on the commands that deal with sentences (navigation such as (backward-sentence) and (forward-sentence), kill such as (kill-sentence) and (backward-kill-sentence). |
| **Toggle local electric quote mode** | `<f11> t m '` | (**electric-quote-local-mode** &optional ARG) | Toggle 'electric-quote-mode' only in this buffer.  Useful to insert nicer-looking quote characters. |
| **Show state of text modes** | `<f11> t m ?` | (**pel-show-text-modes**) | Display the state of the various text modes in the mini buffer. |
| **Text Whitespace Modes** | | The following Emacs command control how whitespace is shown or hidden.  The following commands are also described in the **Text Whitespace** table. | |
| **Toggle Whitespace Mode**<br><br>( Also in: ∑ Whitespace) | `<f11> t w m` | (**whitespace-mode** &optional ARG) | Toggle whitespace visualization (Whitespace mode).<br>With a prefix argument ARG, enable Whitespace mode if ARG is positive, and disable it otherwise.<br>The kind of whitespace visualized is determined by the list variable **whitespace-style**, **whitespace-newline**. |
| **Hide/Show trailing whitespaces**<br>( Also in: ∑ Whitespace) | `<f11> t w T` | (**pel-toggle-show-trailing-whitespace**) | Toggle highlight of the trailing whitespaces in current buffer.<br><br>Toggles the value of the variable **show-trailing-whitespace**. |
| **Hide/Show trailing empty lines**<br>( Also in: ∑ Whitespace) | `<f11> t w e` | (**pel-toggle-indicate-empty-lines**) | Toggle highlight of empty lines.<br><br>Toggles the value of the variable **indicate-empty-lines**. |
| **Toggle individual elements of whitespace-style**<br><br>( Also in: ∑ Whitespace) | `<f11> t w o` | (**whitespace-toggle-options** ARG) | • If local whitespace-mode is off, toggle the option given by ARG and turn on local whitespace-mode.<br>• If local whitespace-mode is on, toggle the option given by ARG and restart local whitespace-mode.<br>The argument, which is a single character and must be typed following the `<f11> t w o`,  can be:<br>• `f`     toggle face visualization<br>• `t`     toggle TAB visualization<br>• `s`     toggle SPACE and HARD SPACE visualization<br>• `r`     toggle trailing blanks visualization<br>• `l`     toggle "long lines" visualization<br>• `L`     toggle "long lines" tail visualization<br>• `n`     toggle NEWLINE visualization<br>• `e`     toggle empty line at bob and/or eob visualization<br>• `C-i`   toggle indentation SPACEs visualization (via 'indent-tabs-mode')<br>• `I`     toggle indentation SPACEs visualization<br>• `i`     toggle indentation TABs visualization<br>• `C-t`   toggle big indentation visualization<br>• `C-a`   toggle SPACEs after TAB visualization (via 'indent-tabs-mode')<br>• `A`     toggle SPACEs after TAB: SPACEs visualization<br>• `a`     toggle SPACEs after TAB: TABs visualization<br>• `C-b`   toggle SPACEs before TAB visualization (via 'indent-tabs-mode')<br>• `B`     toggle SPACEs before TAB: SPACEs visualization<br>• `b`     toggle SPACEs before TAB: TABs visualization<br>• `?`     Show the above list of options. |
| **Mark and Region**<br><br>( Also in: ∑ Marking) | | With most editors when you type over a "selected" region, the text in the selection is automatically replaced by the new text.  By default Emacs does not behave like this; instead it allows typing text while there is an active a marked region.  If you want Emacs behave like other editors and automatically replace the text activate the "*delete-selection-mode*" with the following command. | |
| **Toggles delete selection mode**<br>Also in: ∑ Cut&Paste, Marking) | `<f11> t m d` | (**delete-selection-mode**) | Toggles delete selection-mode on/off.  In delete selection-mode typing a character while a region is active replaces the entire region with what is typed.  By default delete selection-mode is off. |
| **Drawing ASCII in Emacs** | | Emacs provides the picture-mode and artist-mode to draw ASCII-based pictures.  Both are available when Emacs runs in graphics and terminal mode.  However, I have not been able to use artist-mode with the mouse, even with xterm-mouse-mode active: each click just prints an ANSI sequence code. | |
| **Picture Mode** | | Emacs supports the picture mode that allow you to move your cursor freely anywhere inside the window, which greatly simplify creating rectangular shapes for tables or even *drawing* ASCII-art.  This work well in both graphics and terminal mode. | |
| **Enter picture mode**<br><br>(See also: ∑ Drawing) | • `<f11> D p`<br>• `<f11> t p` | (**picture-mode**) | Switch to Picture mode, in which a quarter-plane screen model is used.<br>👉 Very useful to type text in vertical fashion when for example, writing reStructuredText table.<br>• Type `C-c  C-c` to exit picture-mode and return to the mode previously used in the buffer. |
| **Artist Mode** | | Although you can get some commands to work in terminal mode, it's best to use artist-mode when running Emacs in graphics mode. | |
| **Toggle artist mode**<br><br>(See also: ∑ Drawing) | `<f11> D a` | (**artist-mode** &optional ARG) | Toggle Artist mode.<br>• With argument ARG, turn Artist mode on if ARG is positive.<br>• Artist lets you draw lines, squares, rectangles and poly-lines, ellipses and circles with your mouse and/or keyboard. |

## Text Modes — References

| Topic & Link | Notes |
|---|---|
| **GNU Emacs Manual: Text - Words** | |
| **GNU Emacs Manual: Text - Sentences** | |
| **GNU Emacs Manual: Text - Paragraphs** | |
| **GNU Emacs Manual: Text - Quotation Marks** | 2 |
| **GNU Emacs Manual: Modes - Minor Modes** | |
| **GNU Emacs Manual: Programs - Other Features Useful for Editing Programs** | |
| **GNU Info Manual: Getting Started - Invisible text in Emacs Info** | |