

File & Directory Local Variables

Operation	Keystroke	Function	Note
<a href="#">Emacs Variables</a>	<p>Emacs supports specifying Emacs variable values. This can be quite useful to control Emacs behaviour for specified file, file types or files or file types inside a specific location.</p> <ul style="list-style-type: none"><li>For example you can:<ul style="list-style-type: none"><li>identify the tab-width for a specific file, or</li><li>the imenu-generic-expression pattern for a specific type of file, extending what is already available to match the DSL you are using inside a specific directory.</li></ul></li></ul> <p>The variables can be specified using 3 methods:</p> <ul style="list-style-type: none"><li><b>Method 1: File Variables.</b> Written inside the file to which the variables apply, using file variables written as special comment at the top or at the bottom of the file.</li><li><b>Method 2: Per-Directory Local Variables.</b> Written inside a special Emacs-specific file (.dir-local.el) to control the value of variables inside all or specified files in the directory or directory tree.</li><li><b>Method 3:</b> With Emacs Lisp code that identifies class of variables for specific file types, files in a directory tree. See:<ul style="list-style-type: none"><li><a href="#">Using Directory Local Variable classes for individual files @ Emacs Wiki</a></li><li><a href="#">dir-locals-set-class-variables and dir-locals-set-directory-class @ Emacs Manual in Per-Directory Local Variables</a></li></ul></li><li>This method will be required if you want to use Emacs file or directory variables for a project where those are not accepted. You can put the required code inside your init.el file. Currently PEL does not support configuring it using customization.</li></ul>		
Open this PDF file. See also: <a href="#">🔗 Help/Info</a>	<f11> f v <f1>	(pel-help-pdf &optional OPEN-WEB-PAGE)	Open the local copy of the <a href="#">🔗 File/Directory Variables</a> PDF file unless a command prefix (like <b>C-u</b> ) was used. In that case it opens the Github-hosted file instead.
<ul style="list-style-type: none"><li><a href="#">Inside a file: File Variables</a></li></ul>	<p>File variables control Emacs variable for the file into which they are specified. Emacs supports 2 ways to write the file variable specification inside a file, these commands support the first of those two ways:</p> <ul style="list-style-type: none"><li>1: the specification of the variable written inside the very first line using the following format: <code>-*- mode: modename; var: value; ... -*-</code><ul style="list-style-type: none"><li>For example, inside a Lisp file: <code>;; -*- mode: Lisp; fill-column: 75; comment-column: 50; -*-</code></li></ul></li><li>2: at the end of a file with a line containing the string 'Local Variables:', and ends with a line containing the string 'End:'.<ul style="list-style-type: none"><li>For example, in a C file:</li></ul></li></ul> <pre>/* Local Variables: */ /* mode: c */ /* comment-column: 0 */ /* End: */</pre> <p>The following commands can be used to add, modify and remove file variable specification on the first line of a file.</p>		
Add a file variable	<f11> f v =	(add-file-local-variable-prop-line VARIABLE VALUE &optional INTERACTIVE)	Add file-local VARIABLE with its VALUE to the -*- line. <ul style="list-style-type: none"><li>This command prompts for variable name and value.</li><li>It deletes all existing settings of VARIABLE (except 'mode' and 'eval') and adds a new file-local VARIABLE with VALUE to the -*- line.</li><li>If there is no -*- line at the beginning of the current file buffer then this function adds it.</li></ul>
Delete a file variable	<f11> f v -	(delete-file-local-variable-prop-line VARIABLE &optional INTERACTIVE)	Delete all settings of file-local VARIABLE from the -*- line.
Copy all directory file variables	<f11> f v c	(copy-dir-locals-to-file-locals-prop-line)	Copy directory-local variables to the -*- line.
<ul style="list-style-type: none"><li><a href="#">Per-Directory Local Variables</a></li></ul>	<p>Emacs supports directory local variables that can apply to all files inside a given directory and its sub-directory tree.</p> <ul style="list-style-type: none"><li>The variables can be restricted by major modes and sub-directories.</li><li>These files are stored inside a file named <b>.dir-locals.el</b> stored in the directory it controls.</li><li>File local variables override directory local variables, so main settings may be stored inside the directory and can then be fined-tuned inside files.</li><li>Normally directory local variables are stored inside a file named <b>.dir-locals.el</b>. It can also be stored inside the file <b>.dir-locals-2.el</b>. Emacs loads them both, and loads the -2 file after the other. That way you can put the .dir-locals.el under version control and use the other to make personal or temporary customizations.</li></ul> <p>The following commands can be used to create and modify the content of the <b>.dir-locals.el</b> file located in the <b>current</b> directory.</p> <p>👉 If you are familiar with Emacs Lisp it is probably easier to edit the .dir-locals.el file directly, following the format described in the <a href="#">Emacs Manual Directory Local Variables</a> section and to be aware of the security concerns and the ability to force customization of settings that may be identified as unsafe by reading the <a href="#">Safety of File Variables</a> section. You may want to customize the value of <b>safe-local-variable-values</b> to allow some of your values. In any case, it's best to check the code in .dir-locals.el and to check the value of safe-local-variable-values.</p> <p>⚠️ Directory variables are loaded <b>after</b> the major mode and its hooks are executed, preventing mode hooks to see the variables.</p> <p>📖 Because Emacs <b>normal-mode</b> function calls (set-auto-mode) first and later calls (hack-local-variables) if at all. <b>Two techniques</b> exists to circumvent this problem, and PEL uses them to allow controlling important variables like <b>imenu-generic-expression</b> the controls the parsing to extract imenu items.</p>		
Add a directory variable	<f11> f v D =	(add-dir-local-variable MODE VARIABLE VALUE)	Add directory-local VARIABLE with its VALUE and MODE to .dir-locals.el.
Delete a directory variable	<f11> f v D -	(delete-dir-local-variable MODE VARIABLE)	Delete all MODE settings of file-local VARIABLE from .dir-locals.el.
Copy file local variable to directory .dir-locals.el	<f11> f v D C	(copy-file-locals-to-dir-locals)	Copy file-local variables to .dir-locals.el.
Others	The following commands can be used to see and control the current directory.		
List/change buffer's current default directory	M-x cd RET	(cd DIR)	Prompt: shows current directory and allows to change the buffer's default directory to a new DIRectory. <ul style="list-style-type: none"><li>If your environment includes a 'CDPATH' variable, try each one of that list of directories (separated by occurrences of 'path-separator') when resolving a relative directory name.</li><li>The path separator is colon in GNU and GNU-like systems.</li></ul>

File & Directory Local Variables — References

Topic & Link	Description
<a href="#">GNU Emacs Manual: File Variables</a>	Describes the concept of Emacs file variables
<a href="#">GNU Emacs: Safety of File Variables</a>	Describes the concept of Emacs directory local variables
<a href="#">GNU Emacs: Per-Directory Local Variables</a>	Describes the safety issues related to Emacs file variables and Emacs way of dealing with safety.
<a href="#">Local Variables @ Emacs Wiki</a>	Short and sweet overview of the concept with description of the interaction with mode hooks. Must read for Emacs Lisp dev.