


# Emacs support for Gleam

Description	Keystroke	Function	Note
<a href="#">Gleam Support</a> <ul style="list-style-type: none"> <li>File associations</li> <li>See also: <a href="#">⌘ Speedbar</a></li> </ul>	Gleam is an experimental functional static-type checking language for the Erlang BEAM. Its support is very basic, it is not yet fully implemented nor fully documented. This is on my todo list. <ul style="list-style-type: none"> <li>Requires the <a href="#">gleam-mode</a> file <a href="#">📄</a> PEL installs it in the utils directory when <a href="#">pel-use-gleam</a> user-option is set to <a href="#">t</a>.</li> <li>PEL associates the files with the .gleam file extension with gleam-mode.</li> <li>PEL activates <a href="#">⌘ Speedbar</a> support for the Gleam files when <a href="#">pel-use-speedbar</a> user-option is on (set to <a href="#">t</a>).</li> <li> However the imenu support is not yet implemented, therefore identification of functions and other elements is not yet done.</li> </ul>		
Open this PDF file. See also: <a href="#">⌘ Help/Info</a>	<div>&lt;f1&gt; SPC M-G &lt;f1&gt;</div> <div>&lt;f12&gt; &lt;f1&gt;</div>	(pel-help-pdf &optional OPEN-WEB-PAGE)	Open the <b>⌘I - Gleam</b> local PDF. If the prefix argument (like <b>C-u</b> or <b>M--</b> ) is used, then it opens the remote GitHub hosted raw PDF instead. If the <a href="#">pel-flip-help-pdf-arg</a> user-option is set it's the other way around.
<a href="#">⌘ Customize</a> PEL Gleam support	<div>&lt;f11&gt; SPC M-G &lt;f2&gt;</div> <div>&lt;f12&gt; &lt;f2&gt;</div>	(pel-customize-pel &optional OTHER-WINDOW)	Customize PEL Gleam support. <ul style="list-style-type: none"> <li>If OTHER-WINDOW is non-nil (use <b>C-u</b>), display in another window.</li> </ul>

## Emacs & Gleam— References

Document	Notes
<a href="#">Gleam home</a>	
<a href="#">gleam implementation @ Github</a>	
<a href="#">Interview with Gleam creator</a>	
<a href="#">gleam-mode - Emacs support</a>	Very early code.