

# 🚧 Emacs support for Eiffel 🚧

Description	Keystroke	Function	Note
<b>Eiffel Editing</b>  Emacs does not provide any built-in mode for Eiffel programming language. ⚠️ To activate support for Eiffel support on PEL, the <code>pel-use-eiffel</code> user-option must be turned on (set to t). • When <code>pel-use-eiffel</code> is turned on the <code>&lt;f11&gt; SPC C-e</code> prefix is made available, corresponding to the <code>&lt;f12&gt;</code> prefix inside Eiffel buffers.  PEL provides support for Eiffel via: 📦 The <code>eiffel-mode</code> external package is required. ⚡ PEL installs and activates it when the <code>pel-use-eiffel-mode</code> or the <code>pel-use-eiffel</code> user-options is turned on. • 🚧 Note that the <code>eiffel-mode</code> package is a fork of an old Xemacs/Emacs package that does not fully work. I started updating the code but it's a low priority for me at the time I'm writing this.  Eiffel files use the <code>.e</code> file extension. PEL adds Eiffel support for <code>Speedbar</code> when the <code>pel-use-speedbar</code> user-option is turned on. 🚧 However, <code>eiffel-mode</code> itself does not support imenu well and therefore the speedbar can only show the name of the files.  🚧 PEL support for Eiffel is not complete. More commands should be provided and documented. and investigation is required.			
Last updated on:	2025-11-05		
<b>Open this PDF file.</b> See also: <a href="#">Help/Info</a>	<code>&lt;f11&gt; SPC C-e &lt;f1&gt;</code> <code>&lt;f12&gt; &lt;f1&gt;</code>	( <code>pel-help-pdf</code> &optional OPEN-WEB-PAGE)	Open the <code>! - Eiffel</code> local PDF. If the prefix argument (like <code>C-u</code> or <code>M--</code> ) is used, then it opens the remote GitHub hosted raw PDF instead. If the <code>pel-flip-help-pdf-arg</code> user-option is set it's the other way around.
<a href="#">Customize</a> PEL Eiffel support	<code>&lt;f11&gt; SPC C-e &lt;f2&gt;</code> <code>&lt;f12&gt; &lt;f2&gt;</code>	( <code>pel-customize-pel</code> &optional OTHER-WINDOW)	Customize PEL Eiffel support. • If OTHER-WINDOW is non-nil (use <code>C-u</code> ), display in another window.
<a href="#">Customize</a> Emacs Eiffel support	<code>&lt;f11&gt; SPC C-e &lt;f3&gt;</code> <code>&lt;f12&gt; &lt;f3&gt;</code>	( <code>pel-customize-library</code> &optional OTHER-WINDOW)	Customize Emacs Eiffel support: eiffel • If OTHER-WINDOW is non-nil (use <code>C-u</code> ), display in another window.
<b>Comments</b>  Toggle display of comments in buffer or active region See also: <a href="#">Comments</a>	<code>&lt;f11&gt; ; ;</code>	( <code>hide/show-comments-toggle</code> &optional START END)	Toggle hiding/showing of comments in the active region or whole buffer. • If the region is active then toggle in the region. Otherwise, in the whole buffer. 📦 This requires the <code>hide-commt.el</code> package (see <a href="#">Comments</a> ). ⚡ PEL activates it when the <code>pel-use-hide-commt</code> user option is t.

## Emacs & Eiffel – References

Document	Notes
The Eiffel programming language	<ul style="list-style-type: none"> <li><a href="#">Eiffel @ Wikipedia</a></li> <li><a href="#">Eiffel home</a></li> <li><a href="#">Eiffel Software .com</a></li> </ul>
	<ul style="list-style-type: none"> <li><a href="#">Eiffel documentation</a></li> <li><a href="#">Eiffel Libraries</a></li> <li><a href="#">Awesome Eiffel</a></li> </ul>
	• <a href="#">Object-Oriented Software Construction, 2nd edition, Bertrand Meyer 1997, online, released by the author.</a>