

## 🚧 Emacs support for Dart 🚧

Description	Keystroke	Function	Note
<b>Dart Editing</b>	<p>Emacs does not provide any built-in mode for <a href="#">Dart programming language</a> programming language. PEL provides support for Dart via the following external packages activated by the <code>pel-use-dart</code> user-option set to the following values:</p> <p>📦 <a href="#">dart-mode</a>    🗨️ when <code>pel-use-dart</code> is set to <code>t</code>.    <a href="#">See 📖 Tree Sitter and 🗨️ Tree-sitter.</a></p> <p>📦 <a href="#">dart-ts-mode</a>    🗨️ when <code>pel-use-dart</code> is set to <code>with-tree-sitter</code>, <b>Emacs &gt;= 30</b> and <code>pel-use-tree-sitter</code> is <code>t</code>. Best choice.</p> <p>PEL:</p> <ul style="list-style-type: none"> <li>Promotes the use of <a href="#">dart-ts-mode</a> for Emacs &gt;= 30.</li> <li>Associates the <code>.dart</code> file extensions with the Dart major mode currently used.</li> <li>🗨️ <a href="#">iMenu</a> is supported by <a href="#">dart-ts-mode</a> but not by <a href="#">dart-mode</a>.</li> <li>Adds Dart support for 📖 <a href="#">Speedbar</a> for <a href="#">dart-ts-mode</a> when the <code>pel-use-speedbar</code> user-option is turned on.</li> <li>Activates PEL functions keys under the <code>&lt;f12&gt;</code> prefix inside Dart buffers.</li> </ul> <p>The official Dart indentation scheme strictly imposes a 2-space indentation scheme <a href="#">despite people requesting more flexibility</a>.</p> <p>🤖 If this is a problem for you, PEL can help 🤖 🗨️ By setting <code>pel-indent-with-tabs-mode-for-dart</code> to a value between 2 and 8, PEL will automatically activates the <code>pel-indent-with-tabs-mode</code> minor mode for Dart buffers using the selected value as the visual indentation rendering width. You will be able to edit the buffer with the tab-based indentation scheme, with further ability to dynamically change the indentation rendered width with the <code>pel-set-tab-width</code> command (bound to <code>&lt;f11&gt;</code> <code>&lt;tab&gt;</code> <code>w</code>). The file is saved using the original 2-space indentation scheme! See 📖 <a href="#">Indentation</a> for more details.</p>		
🤖 Edit Dart code using wider indentation in buffer, keep 2-space indent in files.	Last updated on: 2025-11-23		
<b>Open this PDF file.</b> See also: 📖 <a href="#">Help/Info</a>	<div> <div>&lt;f11&gt; SPC d &lt;f1&gt;</div> <div>&lt;f12&gt; &lt;f1&gt;</div> </div>	<div> <div>(<a href="#">pel-help-pdf</a> &amp; optional OPEN-WEB-PAGE)</div> </div>	Open the 🗨️ <code>I - Dart</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 Dart support	<div> <div>&lt;f11&gt; SPC d &lt;f2&gt;</div> <div>&lt;f12&gt; &lt;f2&gt;</div> </div>	<div> <div>(<a href="#">pel-customize-pel</a> &amp; optional OTHER-WINDOW)</div> </div>	Customize PEL Dart support. <ul style="list-style-type: none"> <li>If OTHER-WINDOW is non-nil (use <code>C-u</code>), display in another window.</li> </ul>
📖 <a href="#">Customize</a> Emacs Dart support	<div> <div>&lt;f11&gt; SPC d &lt;f3&gt;</div> <div>&lt;f12&gt; &lt;f3&gt;</div> </div>	<div> <div>(<a href="#">pel-customize-library</a> &amp; optional OTHER-WINDOW)</div> </div>	Customize Emacs Dart support: <code>dart</code> , <code>dart-ts</code> <ul style="list-style-type: none"> <li>If OTHER-WINDOW is non-nil (use <code>C-u</code>), display in another window.</li> </ul>
<b>Toggle between classic and Tree-Sitter major mode</b>	PEL supports switching from classic to tree-sitter based mode with the following command. <a href="#">See 📖 Tree Sitter</a>		
	<div> <div>&lt;f11&gt; C-t C-t</div> </div>	<div> <div>(<a href="#">pel-treesit-toggle-mode</a>)</div> </div>	Toggle the major mode between the classic mode and the Tree-Sitter based mode. <ul style="list-style-type: none"> <li>If the other major mode is not available the command signals a user error.</li> </ul> 👉 Use the repeat command (bound to <code>&lt;f5&gt;</code> under PEL) to quickly toggle from the classic to the Tree-Sitter major mode and compare the impact of syntax highlighting.
<b>Show PEL setup information for the major mode.</b>	<div> <div>&lt;f11&gt; ? /</div> <div>&lt;f11&gt; SPC d ?</div> <div>&lt;f12&gt; ?</div> </div>	<div> <div>(<a href="#">pel-mode-setup-info</a> &amp; optional APPEND)</div> <div>(<a href="#">pel-dart-setup-info</a> &amp; optional APPEND)</div> </div>	Display Dart setup information inside a "pel-dart-info" buffer with buttons providing quick access to the customization buffer of each variable shown. The information shown includes the value and interpretation of: <ul style="list-style-type: none"> <li><code>pel-use-dart</code> (whether the classic or tree-sitter based major mode is used).</li> <li>indentation and hard tab control customizable user-options</li> </ul> To append information in the buffer instead of clearing the previous content type any prefix argument (such as <code>C-u</code> ) before the command keystroke.
<b>Comments</b>			
<b>Toggle display of comments in buffer or active region</b> See also: 📖 <a href="#">Comments</a>	<div> <div>&lt;f11&gt; ; ;</div> </div>	<div> <div>(<a href="#">hide/show-comments-toggle</a> &amp; optional START END)</div> </div>	Toggle hiding/showing of comments in the active region or whole buffer. <ul style="list-style-type: none"> <li>If the region is active then toggle in the region. Otherwise, in the whole buffer.</li> </ul> 📦 This requires the <a href="#">hide-comnt.el</a> package (see 📖 <a href="#">Comments</a> ). 🗨️ PEL activates it when the <code>pel-use-hide-comnt</code> user option is <code>t</code> .
<b>Insert, realign, comment/uncomment region</b> See also: 📖 <a href="#">Comments</a> <b>With PEL:</b> Comment the current line with <code>M-0 M-;</code>	<div> <div>M-;</div> </div>	<div> <div>(<a href="#">comment-dwim</a> ARG)</div> <div>(<a href="#">pel-comment-dwim</a> ARG)</div> </div>	Insert or realign comment on current line (or region if a region is active). <ul style="list-style-type: none"> <li>On a single line, the comment is placed <i>after</i> the code.</li> <li><code>C-u M-;</code> executes comment-kill</li> </ul> Same as <code>comment-dwim</code> but comments the current line with a numeric ARG or 0.
<b>Indentation</b>	See also: 📖 <a href="#">Indentation</a>	If you suffer from <a href="#">this issue</a> , you can use PEL or the <a href="#">tab-based-indent</a> to work around the problem.	
<b>Toggle indent-with-tabs-mode</b> 🤖🤖	<div> <div>&lt;f11&gt; &lt;tab&gt; i m</div> </div>	<div> <div>(<a href="#">pel-indent-with-tabs-mode</a> &amp; optional ARG)</div> </div>	Toggle minor mode that automatically changes to tab-based indentation allowing wider rendering via larger tab widths. Save files without the tabs, with original indent scheme.
A PEL independent minor mode exists: <a href="#">tab-based-indent</a> package; it provides the same functionality but outside of PEL.			

## Emacs & Dart — References

Document	Notes
The Dart programming language	<ul style="list-style-type: none"> <li><a href="#">Dart @ Wikipedia</a></li> <li><a href="#">Dart home</a></li> <li><a href="#">Dart @ Github</a></li> <li></li> </ul>
Dart Documentation	<ul style="list-style-type: none"> <li><a href="#">Introduction to Dart</a></li> <li><a href="#">Effective Dart</a></li> <li><a href="#">Dart Libraries</a></li> <li><a href="#">Dart SDK</a></li> <li><a href="#">Asynchronous programming: futures, async, await</a></li> <li><a href="#">Packages</a></li> </ul>
Dart Projects	<ul style="list-style-type: none"> <li><a href="#">Awesome Dart</a></li> </ul>