↖ Topic Index

🚧 Emacs support for Dart 🚧

Description	Keystroke	Function	Note
Dart Editing	Emacs does not provide any built-in mode for Dart programming language programming language. PEL provides support for Dart via the following external packages activated by the pel-use-dart user-option set to the following values: 📦 **dart-mode** when **pel-use-dart** is set to **t**. See ⅀ **Tree Sitter** and 🚦 **Tree-sitter**. 📦 **dart-ts-mode** when **pel-use-dart** is set to **with-tree-sitter, Emacs >= 30** and **pel-use-tree-sitter** is **t**. Best choice. PEL: • Promotes the use of **dart-ts-mode** for Emacs >= 30. • Associates the `.dart` file extensions with the Dart major mode currently used. • ⅀**iMenu** is supported by **dart-ts-mode** but not by **dart-mode**. • Adds Dart support for ⅀ **Speedbar** for **dart-ts-mode** when the **pel-use-speedbar** user-option is turned on. • Activates PEL functions keys under the **<f12>** prefix inside Dart buffers.		
😎 Edit Dart code using wider indentation in buffer, keep 2-space indent in files.	The official Dart indentation scheme strictly imposes a 2-space indentation scheme despite people requesting more flexibility. 😰 If this is a problem for you PEL can help 😎 By setting **pel-indent-with-tabs-mode-for-dart** to a value between 2 and 8, PEL will automatically activates the pel-indent-with-tabs-mode 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 **pel-set-tab-width** command (bound to **`<f11> <tab> w`**). The file is saved using the original 2-space indentation scheme! See ⅀ **Indentation** for more details.		
Last updated on:	2025-11-07		
Open this PDF file. See also: ⅀ **Help/Info**	**`<f11> SPC d <f1>`**	(**pel-help-pdf** &optional OPEN-WEB-PAGE)	Open the 𝔓𝔏 **- Dart** local PDF. If the prefix argument (like **`C-u`** or **`M--`**) is used, then it opens the remote GitHub hosted raw PDF instead. If the **pel-flip-help-pdf-arg** user-option is set it's the other way around.
	`<f12> <f1>`		
⅀ **Customize** PEL Dart support	**`<f11> SPC d <f2>`**	(**pel-customize-pel** &optional OTHER-WINDOW)	Customize PEL Dart support. • If OTHER-WINDOW is non-nil (use **`C-u`**), display in another window.
	`<f12> <f2>`		
⅀ **Customize** Emacs Dart support	**`<f11> SPC d <f3>`**	(**pel-customize-library** &optional OTHER-WINDOW)	Customize Emacs Dart support: dart, dart-ts • If OTHER-WINDOW is non-nil (use **`C-u`**), display in another window.
	`<f12> <f3>`		
Toggle between classic and Tree-Sitter major mode	PEL supports switching from classic to tree-sitter based mode with the following command. See ⅀ **Tree Sitter**		
	`<f11> C-t C-t`	(**pel-treesit-toggle-mode**)	Toggle the major mode between the classic mode and the Tree-Sitter based mode. • If the other major mode is not available the command signals a user error. ☝Use the repeat command (bound to **`<f5>`** under PEL) to quickly toggle from the classic to the Tree-Sitter major mode and compare the impact of syntax highlighting.
Show PEL setup information for the major mode.	**`<f11> ? /`**	(**pel-mode-setup-info** &optional APPEND)	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: • pel-use-dart (whether the classic or tree-sitter based major mode is used). • indentation and hard tab control customizable user-options To append information in the buffer instead of clearing the previous content type any prefix argument (such as **`C-u`**) before the command keystroke.
	`<f11> SPC d ?`	(**pel-dart-setup-info** &optional APPEND)	
	`<f12> ?`		
Comments			
Toggle display of comments in buffer or active region See also: ⅀ **Comments**	**`<f11> ; ;`**	(**hide/show-comments-toggle** &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 **hide-comnt.el** package (see ⅀ **Comments**). PEL activates it when the **pel-use-hide-comnt** user option is **t**.
Insert, realign, comment/ uncomment region See also: ⅀ **Comments** **With PEL**: Comment the current line with **`M-0 M-;`**	**`M-;`**	(**comment-dwim** ARG)	Insert or realign comment on current line (or region if a region is active). • On a single line, the comment is placed *after* the code. • **`C-u M-;`** executes comment-kill
		(**pel-comment-dwim** ARG)	Same as **comment-dwim** but comments the current line with a numeric ARG or 0.

Emacs & Dart — References

Document	Notes
The Dart programming language	• **Dart @ Wikipedia** • **Dart home** • **Dart @ Github** •
Dart Documentation	• **Introduction to Dart** • **Effective Dart** • **Dart Libraries** • **Dart SDK** • **Asynchronous programming: futures, async, await** • **Packages**
Dart Projects	• **Awesome Dart**