Emacs Tree Sitter Support 🚧

Operation	Keystroke	Function	Note
Tree-Sitter Support Last updated on:	Emacs supports Tree-Sitter syntax parsing since Emacs 29. • Emacs 30 introduced a breaking change in the way built-in major modes implement tree-sitter support. • Because of that change PEL supports Tree-sitter on **Emacs >= 30** only. PEL provides extra key binding for operations related to tree sitter when Emacs >= 30 is build with tree-sitter support 2025-10-07		
Open this PDF file. See also: ⬚ **Help/Info**	`<f11> C-t <f1>`	**(pel-treesit-help** &optional OPEN-WEB-PAGE)	Open the ⬚ **Tree-Sitter** 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.
⬚ **Customize** PEL tree-sitter control	`<f11> C-t <f2>`	**(pel-treesit-customize** &optional OTHER-WINDOW)	Customize PEL support for: open the pel-pkg-for-tree-sitter customization group buffer in current window. • If OTHER-WINDOW is non-nil (use `C-u`), display in other window.
⬚ **Customize** Emacs tree-sitter control	`<f11> C-t <f3>`	**(pel-treesit-emacs-customize** &optional OTHER-WINDOW)	Customize Emacs tree-sitter support: open the trees it group buffer in current window. • If OTHER-WINDOW is non-nil (use `C-u`), display in other window.
Toggle the tree-sitter explore minor mode.	`<f11> C-t e`	**(treesit-explore-mode** &optional ARG)	Enable exploring the current buffer's syntax tree. • Pops up a window showing the syntax tree of the source in the current buffer in real time. The corresponding node enclosing the text in the active region is highlighted in the explorer window. • This is a minor mode. If called interactively, toggle the 'Treesit-Explore mode' mode. If the prefix argument is positive, enable the mode, and if it is zero or negative, disable the mode. • If called from Lisp, toggle the mode if ARG is 'toggle'. Enable the mode if ARG is nil, omitted, or is a positive number. Disable the mode if ARG is a negative number.

Toggle tree-sitter inspection mode	`<f11> C-t i`	**(treesit-inspect-mode** &optional ARG)	Minor mode that displays in the mode-line the node which starts at point.
			• When this mode is enabled, the mode-line displays
			
			 PARENT FIELD-NAME: (NODE FIELD-NAME: (CHILD (...)))
			
			 where NODE, CHILD, etc, are nodes which begin at point.
			  • PARENT is the parent of NODE. NODE is displayed in bold typeface.
			  • FIELD-NAMEs are field names of NODE and CHILD, etc (see Info node '(elisp)Language Grammar', heading "Field names").
			
			• If no node starts at point, i.e., point is in the middle of a node, then the mode line displays the earliest node that spans point, and its immediate parent.
			
			• This minor mode doesn't create parsers on its own. It uses the first parser in 'treesit-parser-list'.
			• This is a minor mode. If called interactively, toggle the 'Treesit-Inspect mode' mode. If the prefix argument is positive, enable the mode, and if it is zero or negative, disable the mode.

Tree Sitter and Emacs— References

Topic & Link	Notes
Tree Sitter	• Tree-Sitter @ Wikipedia • Tree-Sitter @ home • Tree-Sitter @ Github

GNU Emacs Lisp Manual	• Using Tree Sitter Parser
	  • Tree-Sitter Language Grammar
	  • Retrieving Nodes
	  • Accessing Node Information
	  • Pattern Matching Tree-Sitter Nodes
	  • User-defined "Things" and Navigation
	  • Parsing text in multiple languages
	  • Developing major modes with Tree-Sitter
	  • Tree-Sitter C API Correspondence

Emacs and Tree-Sitter	• Emacs Tree-Sitter @ Github , a collection of tree-sitter Emacs projects
Articles about Emacs and Tree-Sitter	• How to get Started with Tree-Sitter from Mickey Petersen • Tree Sitter and the Complications of Parsing Languages, from Mickey Petersen

Packages using Tree-Sitter	• **combobulate** @ Github
	  • Combobulate: Structured Movement and Editing with Tree-Sitter, from Mickey Petersen
	  • Combobulate @ reddit

Emacs, PEL and Tree-sitter	• Using tree-sitter with Emacs and PEL describes how PEL currently deals with tree-sitter and how to setup the environment for using tree-sitter with Emacs.