









































































































🚦 Tree-Sitter parsers for Emacs 🚧🚧🚧

TreeSitter parsers	Supported by PEL	User-option control	tree-sitter mode	With  iMenu support	With  Speedbar support							
Last updated on: 2025-10-14 See Also:  Tree Sitter	Indicates yes only when explicitly supported by PEL code.	The name and value of PEL user option that control whether Tree-Sitter aware mode is used.	The name of the major mode command that supports the tree-sitter based control.	Whether all commands based on imenu work in tree-sitter mode.	Whether Speedbar support works for the tree-sitter based mode.	 As PEL introduces explicit support for more major mode, new class will be filled. Once enough tree-sitter support is explicitly implemented, I will add explicit support for LSP and then check the support of various features like completion, navigation based on LSP and tree-sitter. I will then add more columns related to these features here and in the  Language Servers table.						
 - Ada  												
 - AppleScript												
APL 												
 - Arc 												
 - awk 												
 - C 												
 - C++  												
Carbon  future 												
 - Chez 												
 - Chibi 												
 - Chicken 												
 - Clojure 												
Common Lisp 												
Crystal 												
 - D  												
Dart 												
 - Eiffel   												
 - Elm  												
 - Elixir   	Yes	pel-use-elixir	elixir-ts-mode	Yes	Yes							
  - Emacs Lisp												
 - Erlang  												
 - Factor   												
 - Forth 												
Fortran 												
 - Gambit 												
 - Gerbil  												
 - GNU Guile 												
 - Gleam	Yes	See note ➡	gleam-ts-mode	Yes	Yes	Note: Gleam is only supported by a Tree-Sitter aware mode. There's no classic mode for Gleam.						
 - Go 	Yes	pel-use-go	go-ts-mode	Yes	Yes							
 - Go go.mod	Yes	pel-use-go	go-mod-ts-mode	Yes	Yes							

TreeSitter parsers	Supported by PEL	User-option control	tree-sitter mode	With  iMenu support	With  Speedbar support							
Groovy 												
<u>⌘</u> - Haskell  (F)												
Haxe 												
<u>⌘</u> - Hy (python) 												
<u>⌘</u> - Janet   												
Java 												
<u>⌘</u> - Javascript 												
<u>⌘</u> - Julia 												
Kotlin 												
<u>⌘</u> - LFE    (A)												
<u>⌘</u> - Lua   												
<u>⌘</u> -Modula												
<u>⌘</u> - NetRexx												
<u>⌘</u> - Nim  												
<u>⌘</u> -Objective-C 												
<u>⌘</u> - OCaml  												
<u>⌘</u> - Odin 												
<u>⌘</u> -Pascal												
<u>⌘</u> - Perl (perl5)												
<u>⌘</u> - Pike   												
<u>⌘</u> - Python    												
<u>⌘</u> - Purescript  (F)												
R     												
<u>⌘</u> - Racket  												
<u>⌘</u> - ReasonML 												
<u>⌘</u> - REXX												
<u>⌘</u> - Ruby												
<u>⌘</u> - Rust 	Yes	pel-use-rust	rust-ts-mode	Yes	Yes							
Scala 												
<u>⌘</u> - Scheme  												
<u>⌘</u> -Seed7    												
<u>⌘</u> -Smalltalk  												
<u>⌘</u> -Swift												
<u>⌘</u> - Tcl  												
<u>⌘</u> - Typescript 												

<u>TreeSitter parsers</u>	Supported by PEL	User-option control	tree-sitter mode	With <u>⌘iMenu</u> support	With <u>⌘ Speedbar</u> support							
<u>⌘ - UNIX Shell</u>												
<u>⌘ - V</u>												
<u>⌘ -Zig</u> 