↖ Topic Index

🚦 Tree-Sitter parsers for Emacs 🚧🚧

TreeSitter parsers	Supported by PEL	User-option control	tree-sitter mode	Language grammar	With ∑iMenu support	With ∑ Speedbar support	Status of the Tree-Sitter aware major mode	Features working in the tree-sitter mode		
Last updated on: 2025-10-19 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. Modes names in black are built-in Emacs.	Name and link to the project providing the language grammar.	Whether all commands based on imenu work in tree-sitter mode.	Whether Speedbar support works for the tree-sitter based mode.	Identify any known problem here. Later this will be expanded to several features			
𝔓𝔩 - Ada 🚧 ϡ🅢	Yes	pel-use-ada	ada-ts-mode	tree-sitter-langs ➡ briot/tree-sitter-ada	Yes	Yes	The ada-ts-mode is a great implementation!	• outline-minor-mode. See ∑ **Outline** • syntax-highligting: 4 levels: controlled by **treesit-font-lock-level** • flexible/precise indentation control with several indentation back-ends: • tree-sitter based (the default), very flexible with 9 customizable user-options. • LSP back-end (a formatter instead of indentation engine)		
𝔓𝔩- AppleScript										
APL 🚧				No grammar exists yet.						
𝔓𝔩 - Arc ⓕⓜ										
𝔓𝔩 - awk ⓓ										
𝔓𝔩 - C 🅢				tree-sitter-langs ➡ tree-sitter/tree-sitter-c						
𝔓𝔩 - C++ 🅞🅢				tree-sitter-langs ➡ tree-sitter/tree-sitter-cpp						
Carbon 🚧 future 🅢										
𝔓𝔩 - Chez ⓕⓜ										
𝔓𝔩 - Chibi ⓕⓜ										
𝔓𝔩 - Chicken ⓕⓜ										
𝔓𝔩 - Clojure ⓕⓜ				tree-sitter-langs ➡ sogaiu/tree-sitter-clojure						
Common Lisp ⓕⓜ				tree-sitter-langs ➡ tree-sitter-grammars/tree-sitter-commonlisp						
Crystal 🚧										
𝔓𝔩 - D ⓘⓕⒶ				tree-sitter-langs ➡ CyberShadow/tree-sitter-d						
Dart 🚧				tree-sitter-langs ➡ UserNobody14/tree-sitter-dart						
𝔓𝔩 - Eiffel 🚧 🅞🅢										
𝔓𝔩 - Elm 🚧 Ⓕ				tree-sitter-langs ➡ elm-tooling/tree-sitter-elm						
𝔓𝔩 - Elixir ⓒⓜⓕⒶ	Yes	pel-use-elixir	elixir-ts-mode	tree-sitter-langs ➡ elixir-lang/tree-sitter-elixir	Yes	Yes	OK			

TreeSitter parsers	Supported by PEL	User-option control	tree-sitter mode	Language grammar	With ⅀iMenu support	With ⅀ Speedbar support	Status of the Tree-Sitter aware major mode	Features working in the tree-sitter mode		
𝔏𝔓𝔏 - Emacs Lisp				tree-sitter-langs ➡ Wilfred/tree-sitter-elisp						
𝔓𝔏 - Erlang ⓒⓕⒶ	Yes	pel-use-erlang	erlang-ts-mode	tree-sitter-langs ➡ WhatsApp/tree-sitter-erlang	Yes	Yes	As of this writing, this is an early version. Fontification only works for comments. Maintainers of erlang-ts-mode would appreciate help.			
𝔓𝔏 - Factor Ⓚⓕⓞⓜ	No: As of Emacs 30.2 there is no factor-ts-mode			No grammar exists yet.			Does not exists yet.			
𝔓𝔏 - Forth Ⓚ	No: As of Emacs 30.2 there is no forth-ts-mode			AlexanderBrevig/tree-sitter-forth	Yes	Yes	Does not exists yet.			
Fortran 🚧				tree-sitter-langs ➡ stadelmanma/tree-sitter-fortran						
𝔓𝔏 - Gambit ⓕⓜ										
𝔓𝔏 - Gerbil ⓕⓜⒶ										
𝔓𝔏 - GNU Guile ⓕⓜ										
𝔓𝔏 - Gleam	Yes	See note ➠	gleam-ts-mode	tree-sitter-langs ➡ tree-sitter-gleam	Yes	Yes	OK	Note: Gleam is only supported by a Tree-Sitter aware mode. There's no classic mode for Gleam.		
𝔓𝔏 - Go 🅢	Yes	pel-use-go	go-ts-mode	tree-sitter-langs ➡ tree-sitter-go	Yes	Yes	OK			
𝔓𝔏 - Go go.mod	Yes	pel-use-go	go-mod-ts-mode	tree-sitter-go-mod	Yes	Yes	OK			
Groovy 🚧				tree-sitter-langs ➡ Decodetalkers/tree-sitter-groovy						
𝔓𝔏 - Haskell Ⓕ				tree-sitter-langs ➡ tree-sitter/tree-sitter-haskell						
Haxe 🚧				tree-sitter-langs ➡ vantreeseba/tree-sitter-haxe						
𝔓𝔏 - Hy *(python)* ⓜ										
𝔓𝔏 - Janet ⓘⓕⓜ				• tree-sitter-langs ➡ sogaiu/tree-sitter-janet-simple , • GrayJack/tree-sitter-janet						
𝔓𝔏 - Java 🚧				tree-sitter-langs ➡ tree-sitter/tree-sitter-java						
𝔓𝔏 - Javascript 🚧				tree-sitter-langs ➡ tree-sitter/tree-sitter-javascript						
𝔓𝔏 - Julia ⓜ				tree-sitter-langs ➡ tree-sitter/tree-sitter-julia						
Kotlin 🚧				tree-sitter-langs ➡ fwcd/tree-sitter-kotlin						
𝔓𝔏 - LFE ⓒⓜⓕⒶ										

TreeSitter parsers	Supported by PEL	User-option control	tree-sitter mode	Language grammar	With ΣiMenu support	With Σ Speedbar support	Status of the Tree-Sitter aware major mode	Features working in the tree-sitter mode		
𝔓𝔩 -Lua ⓕ 🄾 ⓟ	Yes	pel-use-lua	lua-ts-mode	tree-sitter-langs ➡ MunifTanjim/tree-sitter-lua ⚠️	Yes	Yes	• fortification does not work • The tree-sitter-lua project used by tree-sitter-langs seems unmaintained. It should probably use tree-sitter-grammars/tree-sitter-lua			
𝔓𝔩-Modula	No: As of Emacs 30.2 there is no modula-ts-mode									
𝔓𝔩 - NetRexx	No: As of Emacs 30.2 there is no netrexx-ts-mode			No grammar exists yet.			Does not exists yet.			
𝔓𝔩 - Nim ⓜ🅢	No: As of Emacs 30.2 there is no nim-ts-mode implemented yet.			alviss/tree-sitter-nim	No	Yes, but since iMenu is not supported, nothing shows.	Does not exists yet.			
𝔓𝔩-Objective-C 🚧										
𝔓𝔩 - OCaml ⓘⓕ	No: there seems to be several tree-sitter aware major modes for OCaml but PEL does not yet support any as there does not seem to have a clear winner.			tree-sitter-langs ➡ tree-sitter/tree-sitter-ocaml			There seems to have several incomplete implementations: • terrateamio/ocaml-ts-mode • dmitrig/ocaml-ts-mode			
𝔓𝔩 - Odin 🅢										
𝔓𝔩-Pascal	No: As of Emacs 30.2 there is no pascal-ts-mode			tree-sitter-langs ➡ isopod/tree-sitter-pascal			Does not exists yet.			
𝔓𝔩 - Perl (perl5)	No: As of Emacs 30.2 there is no perl-ts-mode implementation that has reached good enough stability.			tree-sitter-langs ➡ tree-sitter-perl/tree-sitter-perl	Yes	Yes	Under development, not yet ready: • sourcehut/pranshu/perl-ts-mode • HaraldJoerg/emacs-perl-ts-mode			
𝔓𝔩 - Pike 𝒅 ⓘ 🄾	No: As of Emacs 30.2 there is no pike-ts-mode			No grammar exists yet.			Does not exists yet.			
𝔓𝔩 - *Python* 𝒅ⓟ🄾🄵				tree-sitter-langs ➡ tree-sitter/tree-sitter-python						
𝔓𝔩 - Purescript 🚧 Ⓕ				tree-sitter-langs ➡ postsolar/tree-sitter-purescript						
R 🚧 🄾 ⓟ ⓕ 🅇 |				tree-sitter-langs ➡ r-lib/tree-sitter-r						
𝔓𝔩 - Racket ⓕⓜ				tree-sitter-langs ➡ 6cdh/tree-sitter-racket						
𝔓𝔩 - ReasonML 🚧										
𝔓𝔩 - REXX	No: As of Emacs 30.2 there is no rexx-ts-mode			No grammar exists yet.			Does not exists yet.			
𝔓𝔩 - Ruby	Yes	pel-use-ruby	ruby-ts-mode	tree-sitter-langs ➡ tree-sitter/tree-sitter-ruby	Yes	Yes	OK			
𝔓𝔩 - Rust 🅢	Yes	pel-use-rust	rust-ts-mode	tree-sitter-langs ➡ tree-sitter/tree-sitter-rust	Yes	Yes	OK			
Scala 🚧				tree-sitter-langs ➡ tree-sitter/tree-sitter-scala						
𝔓𝔩 - Scheme ⓕⓜ				tree-sitter-langs ➡ 6cdh/tree-sitter-scheme						
𝔓𝔩-Seed7 🚧 ⓔ ⓖ ℷ	No: As of Emacs 30.2 there is no seed7-ts-mode implemented yet.			No grammar exists yet.	Yes, for seed7-mode	Yes, for seed7-mode	Does not exists yet.			

TreeSitter parsers	Supported by PEL	User-option control	tree-sitter mode	Language grammar	With ∑iMenu support	With ∑ Speedbar support	Status of the Tree-Sitter aware major mode	Features working in the tree-sitter mode		
𝔓𝔩-Smalltalk 🚧 ⓞ										
𝔓𝔩-Swift				tree-sitter-langs ➡ alex-pinkus/tree-sitter-swift						
𝔓𝔩 - Tcl ⓕⓘ	No: As of Emacs 30.2 there is tcl-ts-mode implemented yet, even though the Tree-Siter grammar exists.			tree-sitter-langs ➡ tree-sitter/tree-sitter-tcl	Yes, for tcl-mode	Yes, for tcl-mode	Does not exists yet.			
𝔓𝔩 - Typescript 🚧				tree-sitter-langs ➡ tree-sitter/tree-sitter-typescript						
𝔓𝔩 - UNIX Shell										
𝔓𝔩 - V	No: As of Emacs 30.2 there is v-ts-mode implemented yet, even though the Tree-Siter grammar exists.			No grammar exists yet.			Does not exists yet. There is nedpals/tree-sitter-v but that does not seemed maintained.			
𝔥𝔡𝔩 - Verilog				tree-sitter-langs ➡ tree-sitter/tree-sitter-verilog						
𝔥𝔡𝔩 - VHDL				tree-sitter-langs ➡ alemuller/tree-sitter-vhdl						
𝔓𝔩 -Zig 🅢	Yes	pel-use-zig	zig-ts-mode	tree-sitter-langs ➡ tree-sitter-grammars/tree-sitter-zig	Yes	Yes	With language grammar of 2025-10-13: • fortification does not work • incomplete indentation control • no format on save like zig-mode			
				maxxino/tree-sitter-zig	Yes	Yes	OK			