

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	Language Server	
Last updated on: 2026-01-17 See Also: <ul style="list-style-type: none">Tree SitterUsing tree-sitter with Emacs and PELList of Tree-Sitter parsers	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 are tree-sitter based. Modes names in black are built-in Emacs. The others have a link	Name and link to the project providing the language grammar. If an entry is required in <code>treesit-load-name-override-list</code> it is identified here.	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			
Ada 🚧 ↗ ⓘ	Yes	pel-use-ada	ada-ts-mode	tree-sitter-langs ➔ briot/tree-sitter-ada	Yes	Yes	The ada-ts-mode is a great implementation!	<ul style="list-style-type: none"> outline-minor-mode. See Outline syntax-highlighting: 4 levels: controlled by treesit-font-lock-level flexible/precise indentation control with several indentation back-ends: <ul style="list-style-type: none"> tree-sitter based (the default), very flexible with 9 customizable user-options. LSP back-end (a formatter instead of indentation engine) 		
Algol	No									
AppleScript				No grammar found yet.						
APL 🚧				No grammar found yet.						
Arc ⓘ	ⓘ			No grammar found yet, but perhaps a scheme syntax could be used.						
awk Ⓞ				Beaglefoot/tree-sitter-awk						
C ⓘ	Yes	pel-use-c		tree-sitter-langs ➔ tree-sitter/tree-sitter-c						
C# 🚧 future										
C++ ⓘ	Yes	pel-use-c++		tree-sitter-langs ➔ tree-sitter/tree-sitter-cpp						
C3 ⓘ	Yes	pel-use-c3	c3-ts-mode	github.com/c3lang/tree-sitter-c3	Yes	Yes	Basic: highlighting, iMenu and Speedbar work fine. No support for navigation commands as of December 2025.			
Carbon 🚧 future ⓘ				No grammar found yet.						
Chez ⓘ	ⓘ			No grammar found yet, but perhaps a scheme syntax could be used.						
Chibi ⓘ	ⓘ			No grammar found yet, but perhaps a scheme syntax could be used.						
Chicken ⓘ	ⓘ			No grammar found yet, but perhaps a scheme syntax could be used.						
Clojure ⓘ	ⓘ			tree-sitter-langs ➔ sogaiu/tree-sitter-clojure						
Common Lisp ⓘ	ⓘ			tree-sitter-langs ➔ tree-sitter-grammars/tree-sitter-commonlisp						
Crystal 🚧				crystal-lang-tools/tree-sitter-crystal						

<u>TreeSitter parsers</u>	Supported by PEL	User-option control	tree-sitter mode	Language grammar	With <u>☰iMenu</u> support	With <u>☰ Speedbar</u> support	Status of the Tree-Sitter aware major mode	Features working in the tree-sitter mode	Language Server	
<u>TreeSitter - D</u> ⓘ f A				tree-sitter-langs ➔ CyberShadow/tree-sitter-d						
<u>TreeSitter - Dart</u>	Yes	pel-use-dart	dart-ts-mode	tree-sitter-langs ➔ UserNobody14/tree-sitter-dart	Yes, only with dart-ts-mode	Yes, only with dart-ts-mode	OK			
<u>TreeSitter - Eiffel</u> ⚡ ⓘ S				No grammar found yet.						
<u>TreeSitter - Elm</u> ⚡ ⓘ F				tree-sitter-langs ➔ elm-tooling/tree-sitter-elm						
<u>TreeSitter - Elixir</u> ⚡ m f A	Yes	pel-use-elixir	elixir-ts-mode	tree-sitter-langs ➔ elixir-lang/tree-sitter-elixir	Yes	Yes	OK			
<u>TreeSitter - Emacs Lisp</u>				tree-sitter-langs ➔ Wilfred/tree-sitter-elisp						
<u>TreeSitter - Erlang</u> ⚡ f A	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.			
<u>TreeSitter - Factor</u> ⓘ f @ M	No: As of Emacs 30.2 there is no factor-ts-mode			No grammar found yet.				Nothing found yet.		
<u>TreeSitter - Forth</u> ⓘ	No: As of Emacs 30.2 there is no forth-ts-mode			AlexanderBrevig/tree-sitter-forth	Yes	Yes		Nothing found yet.		
Fortran ⚡				tree-sitter-langs ➔ stadelmanma/tree-sitter-fortran						
<u>TreeSitter - Gambit</u> ⓘ M				No grammar found yet, but perhaps a scheme syntax could be used.						
<u>TreeSitter - Gerbil</u> f m A				No grammar found yet, but perhaps a scheme syntax could be used.						
<u>TreeSitter - GNU Guile</u> f M				No grammar found yet, but perhaps a scheme syntax could be used.						
<u>TreeSitter - Gleam</u>	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.		
<u>TreeSitter - Go</u> ⓘ	Yes	pel-use-go	go-ts-mode	tree-sitter-langs ➔ tree-sitter-go	Yes	Yes	OK			
<u>TreeSitter - Go</u> 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						
<u>TreeSitter - Haskell</u> ⓘ				tree-sitter-langs ➔ tree-sitter/tree-sitter-haskell						
Haxe ⚡				tree-sitter-langs ➔ vantreeeseba/tree-sitter-haxe						
<u>TreeSitter - Hy</u> (python) ⓘ				No grammar found yet.						
<u>TreeSitter - Janet</u> ⓘ f M				<ul style="list-style-type: none"> tree-sitter-langs ➔ sogaiu/tree-sitter-janet-simple, GrayJack/tree-sitter-janet 						
<u>TreeSitter - Java</u> ⚡										

<u>TreeSitter parsers</u>	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	Language Server	
- Javascript	Yes	pel-use-js	js-ts-mode	tree-sitter-langs ➔ tree-sitter/tree-sitter-javascript	Yes	Yes	OK			
- Julia				tree-sitter-langs ➔ tree-sitter/tree-sitter-julia						
Kotlin				tree-sitter-langs ➔ fwcd/tree-sitter-kotlin						
- LFE				No grammar found yet.						
- Lua	Yes	pel-use-lua	lua-ts-mode	tree-sitter-grammars/tree-sitter-lua	Yes	Yes	OK			
				tree-sitter-langs ➔ MunifTanjim/tree-sitter-lua !	Yes	Yes	<ul style="list-style-type: none"> fortification does not work The tree-sitter-lua project used by tree-sitter-langs seems unmaintained. 			
- Modula				No: As of Emacs 30.2 there is no modula-ts-mode	No grammar found yet.					
- NetRexx				No: As of Emacs 30.2 there is no netrexx-ts-mode	No grammar found yet.			Nothing found yet.		
- 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.	Nothing found yet.		
- Objective-C				No: As of Emacs 30.2 there is no known objc-ts-mode implemented yet.	<ul style="list-style-type: none"> tree-sitter-grammars/tree-sitter-objc merico-dev/tree-sitter-objc 					
- 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: <ul style="list-style-type: none"> terrateamio/ocaml-ts-mode dmitrig/ocaml-ts-mode 		
- Odin				Sampie159/odin-ts-mode	tree-sitter-grammars/tree-sitter-odin				DanielGavin/ols	
- Pascal				No: As of Emacs 30.2 there is no pascal-ts-mode	tree-sitter-langs ➔ isopod/tree-sitter-pascal			Nothing found yet.		
- Perl				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: <ul style="list-style-type: none"> sourcehut/pranshu/perl-ts-mode HaraldJoerg/emacs-perl-ts-mode 		
- Pike				No: As of Emacs 30.2 there is no pike-ts-mode	No grammar found yet.			Nothing found yet.		
- Python					tree-sitter-langs ➔ tree-sitter/tree-sitter-python					
- Purescript					tree-sitter-langs ➔ postsolar/tree-sitter-purescript					
R					tree-sitter-langs ➔ r-lib/tree-sitter-r					
- Racket					tree-sitter-langs ➔ 6cdh/tree-sitter-racket					
- ReasonML					reasonml-editor/tree-sitter-reason			Nothing found yet.		

<u>TreeSitter parsers</u>	Supported by PEL	User-option control	tree-sitter mode	Language grammar	With <u>☰iMenu</u> support	With <u>☰ Speedbar</u> support	Status of the Tree-Sitter aware major mode	Features working in the tree-sitter mode	Language Server	
<u>REXX</u>			No: As of Emacs 30.2 there is no rex-ts-mode	No grammar found yet.			Nothing found yet.			
<u>Ruby</u>	Yes	pel-use-ruby	ruby-ts-mode	tree-sitter-langs ➔ tree-sitter/tree-sitter-ruby	Yes	Yes	OK			
<u>Rust</u> ⓘ	Yes	pel-use-rust	rust-ts-mode	tree-sitter-langs ➔ tree-sitter/tree-sitter-rust	Yes	Yes	OK			
<u>Scala</u> ⚡				tree-sitter-langs ➔ tree-sitter/tree-sitter-scala						
<u>Scheme</u> ⓘ ⓘ				tree-sitter-langs ➔ 6cdh/tree-sitter-scheme						
<u>Seed7</u> ⚡ ⓘ ⓘ ↗	No: As of Emacs 30.2 there is no seed7-ts-mode implemented yet.			No grammar found yet.	Yes, for seed7-mode	Yes, for seed7-mode	Nothing found yet.			
<u>Smalltalk</u> ⚡ ⓘ				No grammar found yet.						
<u>Swift</u>				tree-sitter-langs ➔ alex-pinkus/tree-sitter-swift						
<u>Tcl</u> ⓘ ⓘ	No: As of Emacs 30.2 there is tcl-ts-mode implemented yet, even though the Tree-Sitter grammar exists.			tree-sitter-langs ➔ tree-sitter/tree-sitter-tcl	Yes, for tcl-mode	Yes, for tcl-mode	Nothing found yet.			
<u>TypeScript</u> ⚡				tree-sitter-langs ➔ tree-sitter/tree-sitter-typescript						
<u>UNIX Shell</u>										
<u>V</u>	No: As of Emacs 30.2 there is v-ts-mode implemented yet, even though the Tree-Sitter grammar exists.			No grammar found yet.			Nothing complete found yet. There is nedpals/tree-sitter-v but that does not seem maintained.			
<u>Verilog</u>				tree-sitter-langs ➔ tree-sitter/tree-sitter-verilog						
<u>VHDL</u>				tree-sitter-langs ➔ alemuller/tree-sitter-vhdl						
<u>Zig</u> ⓘ	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: <ul style="list-style-type: none">• fortification does not work• incomplete indentation control• no format on save like zig-mode			
				maxxino/tree-sitter-zig	Yes	Yes	OK			