Emacs support for Perl

Description	Kayatraka	Eunstion	Note			
Description	Keystroke	Function	Note			
Perl Editing See: Perl 5 Syntax		Emacs provides two major modes for Perl: perl-mode (Emacs default, simpler mode) and operl-mode . The HaraldJoerg/cperl-mode external package has support for new Perl language features.				
 Activate Perl ⇒ Select major mode 	PEL activates Perl support with the pel-use-perl user-options. When turned on:					
•	 PEL provides supports all of them: pel-perl-mode selects the mode, using paraldJoerg/cperl-mode by default: it best supports Perl and perltidy. After using HaraldJoerg/cperl-mode if you want to revert to Emacs' own cperl-mode, remove the cperl-mode. The perl-mode if you want to revert to Emacs' own cperl-mode. 					
<u></u>	PEL activates minor m	nodes for the perl major mode as specified b	by the pel-perl-activates-minor-modes user-option.			
• <u>∑ Speedbar</u> . ➡		s on, speedbar support for Perl is activated. oth perl-mode and cperl-mode is controlle				
• <u>Indentation</u> control	pel-general-perl-inde	ent-level: it defaults to 4. It is applied to per	rl-indent-level and cperl-indent-level as well as tab-width.			
•	The use of hard tabs is c • pel-general-perl-use-	ontrolled by: - tabs : it defaults to nil, forcing use of space:	s. It is applied to indent-tabs-mode.			
PEL Perl support			s. Set pel-cperl-show-trailing-whitespace-normally user-option to t to use the trailing-whitespace			
improvements		It this is a red background whitespace. upport for cperl-mode when pel-use-iedit is	on. See <u>∑ Search/Replace</u> .			
		n find Perl files in Perl directories. It supports	s the Perl package :: and ' syntax. 1-mode perl-tidy-ediff with improved command interface (2 commands do everything instead of 3).			
			avigation between compilation-mode error message and code.			
Open this PDF file. See also: Help/Info	<f11> SPC P <f1></f1></f11>	(pel-help-pdf &optional OPEN-WEB-PAGE)	Open the <u>\$1 - Perl</u> local PDF. With prefix argument (like C-u or M) opens the remote GitHub			
oce also. <u>z rielp/iiilo</u>	<f12> <f1></f1></f12>	Tracj	hosted raw PDF instead. If the pel-flip-help-pdf-arg user-option is set it's the other way around.			
<u>S</u> Customize PEL Perl support	<f11> SPC P <f2></f2></f11>	(pel-customize-pel &optional OTHER-WINDOW)	Customize PEL Perl support. • If OTHER-WINDOW is non-nil (use C-u), display in another window.			
зарроге	<f12> <f2></f2></f12>	Window,	ii Official Window is not all disc e-a), display in another window.			
<u>S</u> Customize Emacs Perl support	<f11> SPC P <f3></f3></f11>	(pel-customize-library &optional OTHER-WINDOW)	Customize Emacs Perl support: perl, cperl, electricity, perl-repl. • If OTHER-WINDOW is non-nil (use C-u), display in another window.			
ren support	<f12> <f3></f3></f12>	OTTEN WINDOW)	in Official vindow is nothin (use c-u), display in another window.			
Show perl-mode status	<f12> <f4> ?</f4></f12>	(pel-perl-show-status &optional APPEND)	Show current buffer 'cperl-mode' status in specialized *pel-perl-info* buffer. • Clear previous buffer content (and move point to top of buffer) unless optional APPEND argument is			
			non-nil,in which case it appends to the previous report and move point to the end of buffer.			
PCRE support			e Perl Compatible regular expression search and operations.			
Perl Doc			ort provides access to the Perl documentation available on your system.			
		n use man instead of peridoc: for example web page provides the complete information	e, man perlintro provides the same information as given by the perldoc perlintro command. n on-line:			
	• Overview	• Operators Modules	• <u>History</u>			
	Tutorials Reference Manual	<u>Functions</u> <u>Utilities</u><u>Variables</u>	 Internals and C Language Interface Language-specific Miscellaneous Platform-specific 			
Show command info	C-c C-h F	(cperl-info-on-command COMMAND)	Show documentation for Perl command COMMAND in other window.			
			 If perl-info buffer is shown in some frame, uses this frame. Customized by setting variables 'cperl-shrink-wrap-info-frame', 'cperl-max-help-size'. 			
peridoc at point	• C-c C-h P	(cperl-perldoc-at-point)	Run a 'perldoc' on the word around point to show information about that Perl word.			
	• <f12> ?</f12>					
	C-c C-h p	(cperl-perldoc WORD &optional SECTION)	Prompt for for (default to word at point). Show information about the selected word with 'perldoc'.			
Show Perl doc	C-c C-h f	(cperl-info-on-current-command)	Show documentation for Perl command at point in other window.			
Help on symbol at	C-c C-h v	(cperl-get-help)	Get one-line docs on the symbol at the point.			
point	The fellowing a common d	and the Clair death of both a test telescope	The data for these docs is a little bit obsolete and may be in fact longer than a line. The data for these docs is a little bit obsolete and may be in fact longer than a line.			
Open file at point (for Perl)	syntax which uses either	:: or ' as path delimiter.	point (the cursor location). It supports extracting Perl package file names from the Perl package name			
∑ File mngt			array and the directories identified by pel-perl-extra-project-root-directories user-option. Instruct Ido to use a file name at point as the basis for the file name to open. This Ido behaviour is			
			you can control it globally or locally with <f11> f M</f11>			
Show searched directory trees	<f12> <f4> .</f4></f12>	(pel-perl-show-source-directories)	Display the list of source directories searched by pel-open-at-point. The list is controlled by Perl's @INC array and the extra list provided by pel-perl-source-directories user-option.			
Open file or web-page	• M- <f6></f6>	(pel-open-at-point &optional N)	Open the file, library or the URL, named at point, with potential line & column #s.			
whose name is at point	• <f11> f .</f11>	(per-open-at-point doptional ty)	Supports glob characters, partial directory path. When multiple files are found it prompts using the			
★★ Command is also	• <u>6y</u>		method selected by pel-prompt-read-method user-option.			
specialized for: • M reStructuredText	This assumes a supplies as	and all the same and the same discontinuous and	will an acid to part to the first series of the series of			
• <u>\$\pi - C</u> , \pi - C++	The command extracts to	he file or directory name, and possibly line a	and is specialized for Perl buffers: it finds Perl files in directories trees identified by Perl's @INC array . Ind column numbers, from text at point and tries to open the file or directory.			
	"	raction works by identifying the beginning &	end of the file/directory/library/URL name string by delimiter characters, one of: tab, newline and:			
• <u>at - UNIX Shell</u> Generic Delimiting	When finding several	I file names, the command lists them and pr	ompts using the method selected by pel-prompt-read-method user-option.			
characters	The default is a very	primitive function implemented by PEL. You	ou can select a more powerful ivy prompting instead.			
	1	 With ivy selected, PEL will automatically set dipel-use-ivy to t and lvy mode will be installed automatically when you restart Emacs. Note that the command shows all files found by the specified search method, it does not only use the first one found. 				
	• Use this to detect potential duplication in package and module files. The command opens the file in the window selected by the following logic controlled by presence of typed numerical profix arguments:					
	The command opens the file in the window selected by the following logic controlled by presence or absence of typed numerical prefix arguments: • Select target window:					
Select target window ▼	• Without argument:					
willdow =	 If no window holds that file, select the target window according to the number of editable windows in frame: if 1, split that window and use the new window, if 2: use the other window, if 3 or more, use the current window. 					
	With prefix numeric	argument N:				
	 (abs N) > 20: there 		erpret the window position from the N value adjusted: N-20 (or N+20 if N is negative)			
N>20 : open the directory ■	 N = 1, 3, 7or abo 		window based on the number of editable windows in frame:			
	if 1 window:if 2 windows:	split that window and use the new use the other window,	window,			
	if 3 or more windows: use the current window. N is: 8: up, 2: down, 4:left, 5:current, 6:right.					
	 N is. 6. up, 2. down, 4.let, 5.current, 6.ngnt. N is 9: force opening the file in the OS associated application (with N=29 or N=-29, open the file's directory with the OS associated application (eg. macOS Finder, Windows Explorer). If this is a URL, open it in the OS default web browser. 					
See function docstring for more info.		r, inexistent or dedicated window is not allow				
Comments	Perl comments start with	#. See <u>E Comments</u> for the generic com	nmands that manage and control comments			
Toggle display of	<f11> ; ;</f11>	(hide/show-comments-toggle	Toggle hiding/showing of comments in the active region or whole buffer.			
comments in buffer or active region		&optional START END)	If the region is active then toggle in the region. Otherwise, in the whole buffer.			
	This requires the hide	e-comnt.el package (see <u>» Comments</u>).	PEL activates it when the pel-use-hide-comnt user option is t .			

Description	<u>Keystroke</u>	Function	<u>Note</u>	
Navigation Commands	Some navigation commands have special behaviour for Perl; they are shown below. They are available in perl-mode and cperl-mode . • There's also several generic navigation commands that work in Perl buffer too. See Navigation for those.			
Move to beginning of function	• C-M-a • <f12> <up></up></f12>	(perl-beginning-of-function &optional ARG)	Move backward to next beginning-of-function, or as far as possible. With argument, repeat that many times; negative args move forward.	
To end of function	• C-M-e • <f12> <down></down></f12>	(perl-end-of-function &optional ARG)	Move forward to next end-of-function. • The end of a function is found by moving forward from the beginning of one.	
Move to next interpolated	C-c C-v	(cperl-next-interpolated-REx &optional SKIP BEG LIMIT)	 With argument, repeat that many times; negative args move backward. Move point to next REx which has interpolated parts. SKIP is a list of possible types to skip, BEG and LIMIT are the starting point and the limit of search (default to point and end of buffer). SKIP may be a number, then it behaves as list of numbers up to SKIP; this semantic may be used as a numeric argument. Types are 0 for / \$rex /o (interpolated once), 1 for /\$rex/ (if \$rex is a result of qr//, this is not a performance hit), t for the rest 	
	C-c C-x	(cperl-next-interpolated-REx-0)	Move point to next REx which has interpolated parts without //o.	
	С-с С-у	(cperl-next-interpolated-REx-1)	Move point to next REx which has interpolated parts without //o. • Skips RExes consisting of one interpolated variable. • Note that skipped RExen are not performance hits.	
Marking Commands	The following marking co	The following marking commands are specialized for Perl. See <u>EMarking</u> for the generic ones that can also be used in Perl buffers.		
Mark function	C-M-h (perl-mark-function) Put mark at end of Perl function, point at beginning			
Indentation Control			See <u>Indentation</u> for the generic ones that can also be used in Perl. erl-mode and cperl-mode customization user-options. Use <f12> <f3> to open customization.</f3></f12>	
Show currently used indentation style	<f12> <f4> s</f4></f12>	(pel-perl-show-style)	Show the name of the currently used indentation style.	
Select new/restore old indentation style	<f12> <f4> <tab></tab></f4></f12>	(pel-perl-set-style &optional PREVIOUS)	Set Perl indentation style to named style. Prompt for indentation style name and apply it. If optional PREVIOUS argument specified, restore the previously used indentation style.	
Indent • perl-mode ➡	<tab></tab>	(perl-indent-command &optional ARG)	Indent Perl code in the active region or current line. In Transient Mark mode, when the region is active, reindent the region. Otherwise with a prefix argument, reindent the current line unconditionally. Otherwise if 'perl-tab-always-indent' is nil and point is not in the indentation area at the beginning of the line, insert a tab. Otherwise, indent the current line. If point was within the indentation area, it is moved to the end of the indentation area. If the line was already indented properly and point was not within the indentation area, and if 'perl-tab-to-comment' is non-nil (the default), then do the first possible action from the following list: 1) delete an empty comment 2) move forward to start of comment, indenting if necessary 3) move forward to end of line 4) create an empty comment 5) move backward to start of comment, indenting if necessary.	
• cperl-mode ►		(cperl-indent-command &optional WHOLE-EXP)	Indent current line as Perl code, or in some cases insert a tab character. • If 'cperl-tab-always-indent' is non-nil (the default), always indent current line. Otherwise, indent the current line only if point is at the left margin or in the line's indentation; otherwise insert a tab. • A numeric argument, regardless of its value, means indent rigidly all the lines of the expression starting after point so that this line becomes properly indented. The relative indentation among the lines of the expression are preserved.	
Indent continued expression	С-м-q	(perl-indent-exp)	Indent each line of the Perl grouping following point.	
• cperl-mode ⇒		(cperl-indent-exp)	Simple variant of indentation of continued-sexp. • Won't indent comment if it starts at 'comment-indent' or looks like continuation of the comment on the previous line. • If 'cperl-indent-region-fix-constructs', will improve spacing on conditional/loop constructs.	
Indent region	C-M-\	(cperl-indent-region START END)	Indents all the lines whose first character is between START and END inclusive. • Won't indent comment if it starts at 'comment-indent' or looks like continuation of the comment on the previous line. • If 'cperl-indent-region-fix-constructs', will improve spacing on conditional/loop constructs.	
Newline and indent • cperl-mode ➡	C-j	(newline-and-indent)	Insert a newline at point, then indent the newly created line. Use it to split a line. Indentation is done using the value of 'indent-line-function' which is set to cperl-indent-line.	
Insert Perl new line	C-c C-j	(cperl-linefeed)	Go to end of line, open a new line and indent appropriately. If in POD, insert appropriate lines.	
Insert matching parens toggle with C-c C-e	foreach (@lin	es) {print; print}	next line with proper indentation, or if you type it inside the inline block of control construct, like form the construct into a multiline and will place you into an appropriately indented blank line. tric-linefeed' documentation. Insert an opening parenthesis or a matching pair of parentheses. Controlled by 'cperl-electric-parens' and 'cperl-hairy' user-options.	
Insert : and indent	•)	(cperl-electric-rparen ARG) (cperl-electric-terminator ARG)	Insert a matching pair of parentheses if marking is active. If not, or if we are not at the end of marking range, would self-insert. Controlled by 'cperl-electric-parens' and 'cperl-hairy' user-options. Insert character and correct line's indentation.	
Insert ; and indent	;	(cperl-electric-semi ARG)	Insert character and correct line's indentation.	
Insert { and indent	{	(cperl-electric-lbrace ARG &optional END)	Insert character, correct line's indentation, correct quoting by space.	
Insert } and indent	}	(cperl-electric-brace ARG &optional ONLY-BEFORE)	Insert character and correct line's indentation. • If ONLY-BEFORE and 'cperl-auto-newline', will insert newline before the place (even in empty line), but not after. If after ")" and the inserted char is "{", insert extra newline before only if 'cperl-extranewline-before-brace'.	
Deleted and possibly untabify	DEL	(cperl-electric-backspace ARG)	Backspace, or remove whitespace around the point inserted by an electric key. • Will untabify if 'cperl-electric-backspace-untabify' is non-nil.	
cperl-mode		ode implementations: Emacs' cperl-mode ammands toggle some of its electric behavior	and p HaraldJoerg/cperl-mode (more complete) activated by the pel-use-perl user-option.	
toggle Perl auto-help	C-c C-h a	(cperl-toggle-autohelp)	Toggle the state of Auto-Help on Perl constructs (put in the message area). • Delay of auto-help controlled by 'cperl-lazy-help-time'. • Delay of auto-help controlled by 'cperl-lazy-help-time'. • Delay of auto-help controlled by 'cperl-lazy-help-time' to an integer value.	
Toggle auto-newline	C-c C-a	(cperl-toggle-auto-newline)	Toggle the state of 'cperl-auto-newline'.	
Toggle electric mode	C-c C-e	(cperl-toggle-electric)	Toggle the state of parentheses doubling in CPerl mode. When typing an opening parens character the closing one is automatically entered.	
Toggle auto-fill mode	C-c C-f • <f11> t f a • <f11> RET</f11></f11>	(auto-fill-mode &optional ARG)	Toggle automatic line breaking (Auto Fill mode). • With a prefix argument, enable Auto Fill mode if the prefix argument is positive, disable it otherwise. • When Auto Fill mode is enabled, inserting a space at a column beyond 'current-fill-column' automatically breaks the line at a previous space.	
	• <f11> RET</f11>			

Toggle keyword expansion	C-c C-k	(cperl-toggle-abbrev)	Toggle the state of automatic keyword expansion in CPerl mode.
Toggle space fix	C-c C-w	(cperl-toggle-construct-fix)	Toggle whether 'indent-region'/'indent-sexp' fix whitespace too.
here-doc support	The following cperl-mode commands operate on Perl here documents .		
Narrows to here-doc	C-c C-n	(cperl-narrow-to-here-doc &optional POS)	Narrows editing region to the HERE-DOC at POS. POS defaults to the point.
Spell-check here-docs	C-c C-d	(cperl-here-doc-spell)	Spell-check HERE-documents in the Perl buffer. • If a region is highlighted, restricts to the region.
Spell-check POD documentation	С-с С-р	(cperl-pod-spell &optional DO-HERES)	Spell-check POD documentation. If invoked with prefix argument, will do HERE-DOCs instead. If a region is highlighted, restricts to the region.
Verify & refactor Perl code	The following commands provide Perl verification and refactoring facilities.		
• perltidy	The following 2 commands run peritidy between areas of the Perl code in the current buffer and the tidied version of that code. All work is done inside Emacs buffers, no file is affected. After generating the tidied code the functions open an ediff session to compare your code and the tidied code, allowing you to decide what to use. Requires HaraldJoerg/cperl-mode activated by the pel-use-perl user-option. The perl-tidy-ediff commands execute the peritidy identified by perl-tidy-command user-option with options identified by the perl-tidy-ediff-args user-option. See Perl::Tidy @meta::cpan for the peritidy list of options, also Online PerlTidy, which provides help on the various options categorized by feature set.		
Run perltidy on current buffer or region. Ediff changes.	<f12> T</f12>	(pel-perl-tidy-ediff)	Run peritidy on current buffer, than start an ediff session, comparing the original source with peritidy output. Error messages are saved in the *peri-tidy-errors* buffer. • If an area is marked, run peritidy on the marked area only.
Run peritidy on current subroutine. Ediff changes. See <u>Narrowing</u>	<f12> t</f12>	(perl-tidy-ediff-sub)	Run the perltidy program on the subroutine that stats before point and start en ediff session to compare the original code with the tidied version. • Error messages are saved in the *perl-tidy-errors* buffer. • The buffer is automatically narrowed to the current function. • When quitting the ediff session it remains narrowed. Use C - x n w to widen the buffer back.
• pericritic	Perform static code anal	ysis with pericritic (which must be installed	separately).
Check code with perioritic	<f12> c</f12>	(pel-perl-critic &optional VERBOSE)	Validate the Perl file visited in current buffer with perloritic. Report error if it's not installed. • With optional VERBOSE prefix argument, print extra information: • Full name of the Policy module that created the violation • Full diagnostic discussion of each Perl Best Practice (PBP) violation. Show errors in compilation-mode buffer in a format that allows navigation.
Other refactoring			
Find/fix missing whitespace code	C-c C-b	(cperl-find-bad-style)	Find places in the buffer where insertion of a whitespace may help. • Prompts user for insertion of spaces. Currently it is tuned to C and Perl syntax.
Refactor: if (A) {B}' ↔ 'B if A;'	C-c C-t	(cperl-invert-if-unless)	Change 'if (A) {B}' into 'B if A;' etc (or visa versa) if possible. • If the cursor is not on the leading keyword of the BLOCK flavor of construct, will assume it is the STATEMENT flavor, so will try to find the appropriate statement modifier.
Lining up Perl Code : X Align	It's often best to line up Perl code vertically, arranging elements as tables, to help reader understand the code intent visually. • This technique is also promoted in <u>Damian Conway's Perl Best Practice book, Vertical Alignment section of chapter 2</u> . See <u>pel-align-words-vertically</u> in <u>E Align</u> .		
Lineup	• C-M- • <f12> </f12>	(cperl-lineup BEG END &optional STEP MINSHIFT)	Lineup construction in a region. Beginning of region should be at the start of a construction. All first occurrences of this construction in the lines that are partially contained in the region are lined up at the same column. MINSHIFT is the minimal amount of space to insert before the construction. STEP is the tabwidth to position constructions. If STEP is nil, 'cperl-lineup-step' will be used (or 'cperl-indent-level', if 'cperl-lineup-step' is nil). Will not move the position at the start to the left.
Testing Perl code	The following command provides a way to test Perl code locally. See Perl 5 Syntax for web-sites that provide a Perl interpreter.		
Start Perl REPL	<f11> z r P</f11>	(perl-repl)	Run a Perl REPL in a *Perl-REPL* buffer.
See: <u>S start Shells/</u> <u>REPLs</u>	<f12> z</f12>		 Requires the <u>perl-repl</u> external package activated by <u>perl-use-perl-repl</u> user-option. The <u>perl-repl-file-path</u> user option specifies the name of the Perl REPL program, which may optionally specify the explicit file path. PEL provides the <u>perl-repl</u> shell script which uses the Perl command line.
		+	

Emacs & Perl — References

Document	Notes
Perl @ Wikipedia	
perl.org	
Learn Perl @ perl.org Perl Style Guide	Perl Tutorial - a gentle introduction to Perl with several examples over a browsable set of pages. Perl Intro - a quick introduction to Perl Online Perl books Beginning Perl
Perl Reference Manuals	 Perl Keywords. Perl Operators. Also see the Perl ABC Operator page: organizes the information in sections (but has some markup typos). Perl Functions
Is Perl still relevant? Most probably.	 What makes Perl relevant in 2022? @ Stackoverflow blog. Perl is dying quick. Could be extinct by 2023. The HFT Guy, 2019. (which includes several invalid points). My point is that Perl was popular, there's a lot of Perl code still being used and it's worth knowing and being able to write and edit Perl code (which was certainly not the first programming language as state by the article). But anyway, the post represents a point of view (and has many people commenting on it). Perl is making a comeback: 5 reasons why it's worth learning. Posted January 6, 2023 by Lucas Rees. Why Perl Didn't Win on outspeaking.com, updated December 21, 2020
Perl and Secure Coding Practice and Tools	Security Issues in Perl Scripts - by Jordan Dimov - Discussion of misused and overlooked features of Perl from the security point of view. SEI CERT Perl Coding Standard @ Carnegie Mellon University The CERT Perl Secure Coding Standard, by David Svoboda , June 25, 2012 perlsec describes Perl security. Tools: The perlcritic script uses Perl::Critic to scan Perl code and provides some listing advices. On some Linux distros you can install it with: sudo dnf install perl-Perl-Critic The perltidy application reformats Perl code to the promoted format. The zarn static analyzer, hosted on Github, is another static analyzer for Perl. (Quite immature as of Oct. 2024).

Document	Notes		
Perl File name extensions	Perl non executable libraries, also used for Perl scripts (but a file with no extension and a shebang line is also fine, and preferable, allowing the invocation of the script without having to type the '.pl'). perl modules. Used by Active State implementation. Identifies executable Perl scripts. Also used elsewhere to distinguish from Prolog (which also uses the .pl file extension). "pls "xs t .pod Plain Old Documentation files, a lightweight markup language used to document Perl code. See also perlpod.		
Perl programs	Perl command line options		
• perl	Perl language interpreter		
peribug / perithanks	Describes how to submit bug report on Perl		
• peridoc	Print Perl Documentation, looking it up in the .pod format embedded in perl installation. Support following options: f: built-in functionsq: FAQ keyword searchv: variablea perl API		
• perlivp	 Perl Installation Verification Procedure: checks Perl installation. Part of perl-level package. 		
• perItidy	PerlTidy @ Wikipedia , PerlTidy Home Pages: @SourceForge @GitHub. Perl::Tidy GitHub repo, Perl::Tidy @meta::cpan Online PerlTidy , which provides help on the various options categorized by feature set.		
perlsec - Perl security			
Getting Perl Modules with CPAN: • <u>CPAN @ Wikipedia</u> • <u>The Zen of Comprehensive Archive Networks</u> • <u>CPAN</u> • <u>Search CPAN — meta::cpan</u> • <u>PAUSE - Perl Authors Upload Server</u>	Command line tools interacting with CPAN: Command line tools i		