

Part Estimage Action for 12 Action	Description	Keystroke	Function	Note	
The furnished and process of the furnished and process of the control of the co	·				
**Because in a wider to subtace the adults of the performance of the p	Activate Perl Select major mode ■	The HaraldJoerg/cperl-mode external package has support for new Perl language features. 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 HaraldJoerg/cperl-mode by default because it has the best support for Perl.			
per bil politic bed acts to a unumer of columns used for indecidents. Per less the bild wide of the first to be formed indecidents occurred to contract to include the contract of columns to income the columns to income the columns to the contract of columns to income the columns to the c	• <u>∑ Indentation</u>	 The <f11> SPC P prefix is made available. In a perl buffer using either major-modes for Perl these commands are accessible via the <f12> key.</f12></f11> It activates the ability to activate minor modes for the perl major mode through the PEL pel-perl-activates-minor-modes user-option. If pel-use-speedbar is on, speedbar support for Perl is activated. 			
See also 2 februs Contaminate PCL Collation (PCL Collation		 perl-indent-level sets the number of columns used for indentation. It defaults to 4. PEL sets tab-width with the same value in Perl buffers so that manual indentation commands use the same number of columns to indent. pel-perl-use-tabs controls whether hard tabs are used for indentation (nil by default). 			
Contention Content C	Open this PDF file. See also: <u>E Help/</u> Info				
Entrace Peri support Comments Toggel designey of Comments (Toggel designey of Toggel designe	© Customize PEL Perl support			···	
Topic display of comments in cashe region or who is suffered as a commentation of the color engine or who is suffered as a color or who is suffered as a color or who is suffered as a color of the color or who is suffered as a color of the	<u>© Customize</u> Emacs Perl support				
Acoptional START END) **If the region is active then region (however, in the whole buffer, require region) **Per to base These commands are provided by Erracis basic Pert support. **The region is active then region (however, in the whole buffer, buffer region) **The region is active then region (however, in the whole buffer, buffer, and the provided by Erracis basic Pert support. **The region is active then region (however, or as or as possible. **Move to segment or or or as or as possible. **Move to segment or	Comments				
Move to beginning C-M-a (perf-beginning-of-function & price of function C-M-e (perf-end-of-function & price of function C-M-e (perf-end-of-function & price of function C-M-b (perf-end-of-function & price of function C-M-b (perf-end-of-function & price of function C-M-b (perf-end-of-function) C-M-b (perf-end-of-function) C-M-b (perf-end-of-function) C-M-c (perf-end-of-function) Put trank at end of Perf burden, perpetitive using move boravaria. Put trank at end of Perf burden, perpetitive using move boravaria. Price of a function burden, perpetitive using move backward. Price of a function burden, perpetitive using move backward. Price of the function burden, perpetitive using move backward. Price of the function, perpetitive using move backward. Price of the function burden, perpetitive using move backward. Price of the function burden, perpetitive using move backward. Price of the function burden, perpetitive using move backward. Price of the function burden, perfect the perpetitive using the function burden burden, perfect the perpetitive using the function burden, perfect the perfect of the perfect	Toggle display of comments in buffer or active region	<f11> ; ;</f11>		• If the region is active then toggle in the region. Otherwise, in the whole buffer. This requires the https://linear.pubmedia.org/https://linear.pubmedia.org/https:/	
Microsoft Intention C-M-e (perf-end-of-function Soptional ARG) (perf-ind-of-function Soptional ARG) (perf-ind-of-function) Put mark at end of Perf function, point at beginning (perf-ind-of-exp) Indent each line of the Perf grouping following point. Value againment, repeat that many times, regalative arge prin once backward. PEL supports 2 operf-model implementations. Emacs' operf-mode and indent any are of Perf code by marking that area and using the tab key. Perf code by marking that area and using the tab key. Perf code by marking that area and using the tab key. Perf code by marking that area and using the tab key. Perf code by marking that area and using the tab key. Perf code by marking that area and using the tab key. Perf code by marking that area and using the tab key. Perf code by marking that area and using the tab key. Perf code to the word area of the perf grouping following points of the word window. Perf code to the word area of the perf code by marking that area and using the tab key. Perf code to the word area of the perf code by marking that area and using the tab key. Perf code to the word area of the perf code by marking that area and using the tab key. Perf code to the word area of the perf code to the word and the perf code to the word area. Perf code to the word area of the perf code to the word area of the perf code to the word area. Perf code to the word area of the perf code to the word area. Perf code to the word area of the perf code to the word area. Perf code to the word area of the perf code to the word area. Perf code to the code to the perf code to the perf	Perl base	These commands are pro	ovided by Emacs basic Perl support.		
The end of a function is found by moving forward from the beginning of one. * Whit argument, repeat that many times, negative sign move backward. * What purpunent, repeat that many times, negative sign move backward. * Put mark at end of Pert function, point at beginning indent * C-N-q perl-indent-axp) Indent each line of the Pert function, point at beginning indent each line of the Pert grouping following point. * Question of the Pert grouping following point to the Pert grouping point to the Pert grouping point point. * Question of the Pert grouping point to show information about that Pert word. * Question of the Pert grouping pertinant point in other window. * Strong of the State of Pert grouping of the State of Pert grouping. * Question of the Question of Pert grouping of the State of Pert grouping and point in other window. * Question of the Question of Pert grouping and point in other window. * Question of the Question of Pert grouping in CPP of mode. * Question of the Question of Pert grouping in CPP of mode. * Question of the Question of Pert grouping in CPP of mode. * Question of the Question of Pert grouping in CPP of mode. * Question of the Question of Pert grouping in CPP of mode. * Question of the Question of Pert grouping in CPP of mode.	Move to beginning of function	С-М-а			
Indent C-Kq (perl-indent-exp) Indent each line of the Perl grouping following point. Stow command COPET-mode implementations: Emacs' cperl-mode and the perfect ode by marking that area and using the tab key. CopeT-indo-on-command COMMAND Info C-c C-h F (opeT-indo-on-command COMMAND) Info C-c C-h P (opeT-indo-on-command COMMAND) Show command COMMAND Info C-c C-h P (opeT-perdoc-at-point) C-c C-h P (opeT-perdoc-at-point) C-c C-h P (opeT-perdoc-at-point) C-c C-h D (opeT-perdoc-WORD Soptional SECITION) Show Perl doc C-c C-h f (opeT-indo-on-courrent-command) Info C-c C-h D (opeT-perdoc-at-point) C-c C-h D (opeT-loggle-auto-newline) C-c C-h D (opeT-loggle-auto-newline) C-c C-a C-a (opeT-loggle-auto-newline) C-c C-a (opeT-loggle-auto-newline) C-c C-a (opeT-loggle-auto-newline) C-c C-b (opeT-loggle-auto-newline) C-c C-b (opeT-loggle-auto-newline) C-c C-c (opeT-loggle-auto-newline) C-c C-	To end of function	С-М-е	(perl-end-of-function &optional ARG)	The end of a function is found by moving forward from the beginning of one.	
performed PEL supports 2 operf-mode implementations: Errace' (operf-mode and implementations) PEL supports 2 operf-mode implementations: Errace' (operf-mode and implementations) PEL supports 2 operf-mode implementations: Errace' (operf-mode (more complete) activated by the pel-tuse-ped user-option show outperformant (operf-mode (more complete) activated by the pel-tuse-ped user-option (operf-perfoce at point) Performance of the performance of the complete is an incomplete activated by the pel-tuse-ped user-option (operf-perfoce at point) Performance of the performance of the complete is an incomplete activated by the pel-tuse-ped user-option (operf-perfoce at point) Performance of the pe	Mark function	C-M-h	(perl-mark-function)	Put mark at end of Perl function, point at beginning	
Show command C-c C-b F (cperf-info-on-command COMMAND) Into Into Into Into Into Into Into Into	Indent	C-M-q	(perl-indent-exp)		
Info C-c C-h P (cperl-peridoc-at-point) Perinfo buffer is shown in some frame, user this frame. Custonized by setting variables 'cperl-shrink-wrap-info-frame', 'cperl-max-help-size'	cperl-mode	PEL supports 2 cperl-me	ode implementations: Emacs' cperl-mode	and MaraldJoerg/cperl-mode (more complete) activated by the pel-use-perl user-option.	
C-c C-h p (cperl-perldoc WORD &potional SECTION) Prompt for for (default to word at point), Show information about the selected word with perildoc'.	Show command info	C-c C-h F	(cperl-info-on-command COMMAND)		
Sisco Per I doc C-c C-h f (cperi-info-on-current-command) Show Word word word word word word word word w	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.	
Help on symbol at point C=C C-h v (operl-get-help) Get one-line docs on the symbol at the point. The data for these docs is a little bit boselet and may be in fact longer than a line. toggie Peri autohelp C=C C-h a (operl-toggle-autohelp) Toggle auto-newline C=C C-a (operl-toggle-auto-newline) Toggle letter C=C C-a (operl-toggle-auto-newline) Toggle the state of port-letter one-newline. Toggle letter Toggle lett		C-c C-h p			
toggle Perl auto- help C-c C-h a (cperl-toggle-autohelp) Toggle auto- help C-c C-h a (cperl-toggle-autohelp) Toggle auto- help C-c C-a (cperl-toggle-autohelp) Toggle auto- newline C-c C-a (cperl-toggle-autonewline) Toggle auto- mode C-c C-e (cperl-toggle-autonewline) Toggle auto- mode C-c C-e (cperl-toggle-autonewline) Toggle auto- mode C-c C-e (cperl-toggle-autonewline) Toggle auto- mode C-c C-f (auto-fill-mode &optional ARG) C-c C-f (auto-fill-mode &optional ARG) Toggle auto- mode C-c C-f (auto-fill-mode &optional ARG) C-c C-f (auto-fill-mode &optional ARG) Toggle auto- mode C-c C-k (cperl-toggle-aborev) C-c C-k (cperl-toggle-aborev) Toggle automatically entered. Toggle automatically mode in the prefix argument is positive, and disable in 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. Toggle space fix C-c C-w (cperl-toggle-construct-fix) Toggle whether 'indent-region'/indent-sexp' fix whitespace too. Find code for whitespace When Auto Fill mode is enabled, inserting a space at a column beyond 'current-fill- column' automatically breaks the line at a previous space. Toggle whether 'indent-region'/indent-sexp' fix whitespace too. Find code for whitespace Whitespace C-c C-b (cperl-find-bad-style) Find code for whitespace Whitespace Whitespace C-c C-d (cperl-here-doc-spell) Spell-check HERE-documents in the Perl buffer. ' if a region is highlighted proprietate lines. This is a convenience replacement for typing carriage return. It places you in the next line with proper indentation, or if you type it inside the inline block of control construct, like foreach (@lines) (print; print) and you are on a boundary of a statement inside braces, it will transform the construct into multiline and will place you into an appropriately indented blank line. I find powed automatically of a statement inside braces, it will transform the construct into multiline and will place you into an	Show Perl doc	C-c C-h f	(cperl-info-on-current-command)	Show documentation for Perl command at point in other window.	
Delay of auto-help controlled by 'cpert-lazy-help-time'. Toggle auto-newline C-c C-a (cpert-toggle-auto-newline) Toggle the state of 'cpert-auto-newline'. Toggle electric C-c C-e (cpert-toggle-electric) Toggle the state of parentheses doubling in CPerl mode. When typing an opening parens character the closing one is anotheratically entered.	Help on symbol at point	C-c C-h v	(cperl-get-help)	· ·	
Toggle electric mode C-c C-e (cperl-toggle-electric) Coggle 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-f (auto-fill-mode & optional ARG) - c fill> t f a column store the closing one is automatically entered. Toggle automatic line breaking (Auto Fill mode). - With a prefix argument, enable Auto Fill mode). - With a prefix argument, enable Auto Fill mode if the prefix argument is positive, and disable it otherwise. - When Auto Fill mode is enabled, inserting a space at a column beyond 'current-fill-column' automatic keyword expansion in CPerl mode. Toggle keyword expansion Toggle space fix C-c C-w (cperl-toggle-construct-fix) Toggle whether 'indent-region'/indent-sexp' fix whitespace too. 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. Spell-check here-does Spell-check here-does C-c C-d (cperl-inefeed) C-c C-j (cperl-linefeed) C-c C-j (cperl-linefeed) C-c C-j (cperl-linefeed) C-c C-j (cperl-linefeed) C-c C-j (cperl-inefeed) Spell-check HERE-documents in the Perl buffer. - If a region is highlighted, restricts to the region. This is a convenience replacement for typing carriage return. It places you in the next line with proper indentation, or if you type it inside the inline block of control construct, like foreach (@lines) (print; print) and you are on a boundary of a statement inside braces, it will transform the construct into insulting and will place you into an appropriately indented blank line. - If you need a usual 'newline-and-indent' behavior, it is on C-j, see documentation on cperl-electric-linefeed'. Narrows to here-doc C-c C-p (cperl-pod-spell &optional Do-HERES) documentation - If invoked with prefix argument, will do HERE-DOCs instead.	toggle Perl auto- help	C-c C-h a	(cperl-toggle-autohelp)	Delay of auto-help controlled by 'cperl-lazy-help-time'.	
Toggle auto-fill mode C-c C-f (auto-fill-mode &optional ARG) C-c C-f (auto-fill-mode &optional ARG) Toggle auto-fill mode if the prefix argument is positive, and disable Auto Fill mode if the prefix argument is positive, and disable for the mode if the prefix argument is positive, and disable for the mode if the prefix argument is positive, and disable for the mode is enabled, inserting a space at a column beyond 'current-fill-column' automatically breaks the line at a previous space. Toggle keyword C-c C-k (cperl-toggle-abbrev) Toggle the state of automatic keyword expansion in CPerl mode. Progle space fix C-c C-b (cperl-toggle-construct-fix) Toggle whether 'indent-region'/indent-sexp' fix whitespace too. Find code for whitespace 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. Spell-check here-doc-spell) Go to end of line, open a new line and indent appropriately. If if a region is highlighted, restricts to the region. C-c C-j (cperl-linefeed) Go to end of line, open a new line and indent appropriately. If in POD, insert appropriate lines. This is a convenience replacement for typing carriage return. It places you in the next line with proper indentation, or if you type it inside the inline block of control construct, like foreach (@lines) {print; print} and you are on a boundary of a statement inside braces, it will transform the construct into multiline and will place you into an appropriately indented blank line. If you need a usual 'newline-and-indent' behavior, it is on C-j, see documentation on 'cperl-electric-linefeed'. Narrows to here-doc Spell-check POD Go-c C-p (cperl-narrow-to-here-doc &optional POS) Spell-check POD documentation. If invoked with prefix argument, will do HERE-DOCs instead.	Toggle auto- newline	C-c C-a	(cperl-toggle-auto-newline)	Toggle the state of 'cperl-auto-newline'.	
* Villa t f a visual to there with a prefix argument, enable Auto Fill mode if the prefix argument is positive, and disable it otherwise. * Villa prefix argument is observed. * Villa prefix argument, enable Auto Fill mode if the prefix argument is positive, and 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. **Toggle keyword** * C-c C-k*** * C-c C-w*** * (cperl-toggle-cabbrev)** * Toggle whether 'indent-region'/indent-sexp' fix whitespace too. * Find code for whitespace** * 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. * Spell-check herBe-documents in the Perl buffer. * If a region is highlighted, restricts to the region. * Find Polo, insert appropriate lines. * This is a convenience replacement for typing carriage return. It places you in the next line with proper indentation, or if you type it inside the inline block of control construct, like foreach (@lines) (print; print) * and you are on a boundary of a statement inside braces, it will transform the construct into a multiline and will place you into an appropriately indented blank line. * If you need a usual 'newline-and-indent' behavior, it is on C-j, see documentation on 'cperl-electric-linefeed'. * Narrows to here-doc * Spell-check POD documentation. * O-c C-p * C-c C-p * (cperl-pod-spell &optional DO-HERES) * Spell-check POD documentation. * If invoked with prefix argument, will do HERE-DOCs instead.		C-c C-e	(cperl-toggle-electric)		
Toggle keyword expansion C-c C-k (cperl-toggle-abbrev) Toggle the state of automatic keyword expansion in CPerl mode. C-c C-w (cperl-toggle-construct-fix) Toggle whether 'indent-region'/indent-sexp' fix whitespace too. 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. Spell-check here-docs Spell-check here-docs C-c C-J (cperl-linefeed) C-c C-J (cperl-linefeed) C-c C-J (cperl-linefeed) C-c C-J (cperl-linefeed) Coc C-J (cperl-linefeed) Narrows to here-doc Coc C-D (cperl-narrow-to-here-doc & optional POS) Coc C-D (cperl-pod-spell & optional DO-HERES)	Toggle auto-fill mode		(auto-fill-mode &optional ARG)	With a prefix argument, enable Auto Fill mode if the prefix argument is positive, and disable it otherwise.	
Toggle space fix C-c C-w (cperl-toggle-construct-fix) Toggle whether 'indent-region'/indent-sexp' fix whitespace too. Find code for whitespace C-c C-b (cperl-find-bad-style) Spell-check heredocs 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. Go to end of line, open a new line and indent appropriately. If in POD, insert appropriate lines. This is a convenience replacement for typing carriage return. It places you in the next line with proper indentation, or if you type it inside the inline block of control construct, like foreach (@lines) (print; print) and you are on a boundary of a statement inside braces, it will transform the construct into a multiline and will place you into an appropriately indented blank line. If you need a usual 'newline-and-indent' behavior, it is on C-j, see documentation on 'cperl-electric-linefeed'. Narrows to heredoc C-c C-n (cperl-narrow-to-here-doc &optional POS) Spell-check POD documentation. C-c C-p (cperl-pod-spell &optional DO-HERES) Spell-check POD documentation. If invoked with prefix argument, will do HERE-DOCs instead.	Toggle keyword	• <f11> RET</f11>	(cperl-toggle-abbrev)	column' automatically breaks the line at a previous space.	
Find code for whitespace 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. Spell-check heredocs 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. Go to end of line, open a new line and indent appropriately. If in POD, insert appropriate lines. This is a convenience replacement for typing carriage return. It places you in the next line with proper indentation, or if you type it inside the inline block of control construct, like foreach (@lines) {print; print} and you are on a boundary of a statement inside braces, it will transform the construct into a multiline and will place you into an appropriately indented blank line. If you need a usual 'newline-and-indent' behavior, it is on C-j, see documentation on 'cperl-electric-linefeed'. Narrows to heredoc C-c C-n (cperl-narrow-to-here-doc &optional POS) Spell-check POD documentation C-c C-p (cperl-pod-spell &optional DO-HERES) Spell-check POD documentation. If invoked with prefix argument, will do HERE-DOCs instead.	expansion		(Total whether for death as 1 1/2 1 1 2 2 2	
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. This is a convenience replacement for typing carriage return. It places you in the next line with proper indentation, or if you type it inside the inline block of control construct, like foreach (@lines) {print; print} and you are on a boundary of a statement inside braces, it will transform the construct into a multiline and will place you into an appropriately indented blank line. If you need a usual 'newline-and-indent' behavior, it is on C-j, see documentation on 'cperl-electric-linefeed'. Narrows to heredoc 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 POD documentation. If invoked with prefix argument, will do HERE-DOCs instead.	Find code for		,	Find places in the buffer where insertion of a whitespace may help.	
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. This is a convenience replacement for typing carriage return. It places you in the next line with proper indentation, or if you type it inside the inline block of control construct, like foreach (@lines) {print; print} and you are on a boundary of a statement inside braces, it will transform the construct into a multiline and will place you into an appropriately indented blank line. If you need a usual 'newline-and-indent' behavior, it is on C-j, see documentation on 'cperl-electric-linefeed'. Narrows to heredoc 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 POD documentation. If invoked with prefix argument, will do HERE-DOCs instead.	Spell-check here-	C-c C-d	(cperi-here-doc-spell)	Spell-check HERE-documents in the Perl buffer.	
with proper indentation, or if you type it inside the inline block of control construct, like foreach (@lines) {print; print} and you are on a boundary of a statement inside braces, it will transform the construct into a multilline and will place you into an appropriately indented blank line. If you need a usual 'newline-and-indent' behavior, it is on C-j, see documentation on 'cperl-electric-linefeed'. Narrows to heredoc 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 POD documentation. If invoked with prefix argument, will do HERE-DOCs instead.	Perl new line	С-с С-ј	(cperl-linefeed)	Go to end of line, open a new line and indent appropriately.	
and you are on a boundary of a statement inside braces, it will transform the construct into a multiline and will place you into an appropriately indented blank line. • If you need a usual 'newline-and-indent' behavior, it is on C-j, see documentation on 'cperl-electric-linefeed'. Narrows to heredoc 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 POD documentation. • If invoked with prefix argument, will do HERE-DOCs instead.					
Narrows to heredoc 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 POD documentation C-c C-p (cperl-pod-spell &optional DO-HERES) Spell-check POD documentation. • If invoked with prefix argument, will do HERE-DOCs instead.				and you are on a boundary of a statement inside braces, it will transform the construct into a multiline and will place you into an appropriately indented blank line.	
Spell-check POD documentation. C-c C-p (cperl-pod-spell &optional DO-HERES) documentation Spell-check POD documentation. • If invoked with prefix argument, will do HERE-DOCs instead.	Narrows to here-	C-c C-n	1 * *		
		С-с С-р	,	If invoked with prefix argument, will do HERE-DOCs instead.	

Description	<u>Keystroke</u>	Function	<u>Note</u>
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.
Move to next interpolated	C-c C-v	(cperl-next-interpolated-REx &optional SKIP BEG LIMIT)	 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.
Insert matching parens	• (• <	(cperl-electric-paren ARG)	Insert an opening parenthesis or a matching pair of parentheses. • Controlled by 'cperl-electric-parens' and 'cperl-hairy' user-options.
(toggle with C-c C-e)	•)	(cperl-electric-rparen 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 : and indent	:	(cperl-electric-terminator ARG)	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-extra-newline-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.
Indent Perl code	TAB	(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.
Newline and indent	C-j	(newline-and-indent)	Insert a newline, then indent according to major mode. • Indentation is done using the value of 'indent-line-function'. In programming language modes, this is the same as TAB.
Indent	C-M-q	(cperl-indent-exp)	Simple variant of indentation of continued-sexp. Will not 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	C-M-\	(cperl-indent-region START END)	Simple variant of indentation of region in CPerl mode. Should be slow. Will not indent comment if it starts at 'comment-indent' or looks like continuation of the comment on the previous line. Indents all the lines whose first character is between START and END inclusive. If 'cperl-indent-region-fix-constructs', will improve spacing on conditional/loop constructs.
Lineup	C-M-	(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.

Emacs & Perl — References

Document	Notes
Perl @ Wikipedia	
<u>perl.org</u>	
Learn Perl @ perl.org	Perl Intro - a quick introduction to Perl