


Perl 5

See also: Perl - Perl <ul style="list-style-type: none">Perl @ Wikipediaperl.orgperldoc browser	Perl Tools	Perl Style Guide. perlcritic script uses Perl::Critic to scan Perl code. The perltidy application reformats Perl code.	
	Learning Perl	<ul style="list-style-type: none">Perl Intro - a quick introduction to PerlOnline Perl books<ul style="list-style-type: none">Beginning Perl	<ul style="list-style-type: none">perl , Perl command line optionsperlvp , perldoc , perlbug / perlthanks perlsec - Perl security
CPAN	<ul style="list-style-type: none">CPAN @ Wikipedia<ul style="list-style-type: none">The Zen of Comprehensive Archive NetworksCPANSearch CPAN — meta::cpanPAUSE - Perl Authors Upload Server		Command line tools interacting with CPAN : <ul style="list-style-type: none">cpan : install on some Linux with: <code>sudo dnf install perl-CPAN</code>cpanplus<code>cpanminus</code> : cpanm : install on some Linux with: <code>sudo dnf install perl-App-cpanminus</code>

Perl 5 Syntax

Perl 5 Operators		Perl has a large number of operators, listed below with their precedence and associativity. <ul style="list-style-type: none">• C Operators missing from Perl : unary &, unary * and (type)• Quote and Quote-like operators : in Perl quotes are operators and they provide various kind of interpolating and pattern matching capabilities.				
Associativity: one of: <ul style="list-style-type: none">• right• left• NA : not associative: cannot use more than one of these operators in sequence.• CH: chained	left	<u>terms and list operators (leftward)</u>				
	left	<u>Arrow Operator:</u> ->				
	NA	<u>Auto-increment and Uato-decrement:</u> ++ --				
	right	<u>Exponentiation:</u> **				
	right	<u>Symbolic Unary Operators:</u> ! ~ -. \ and unary + and -				
	left	<u>Binding operators:</u> =~ !~				
	left	<u>Multiplicative Operators:</u> * / % x				
	left	<u>Additive Operators:</u> + - .				
	left	<u>Shift Operators:</u> << >>				
	NA	<u>named unary operators</u>				
	NA	<u>Class instance Operator:</u> isa				
	CH	<u>Relational Operators:</u> < > <= >= lt gt le ge				
	CH/NA	<u>Equality Operators:</u> == != eq ne <=> cmp ==				
	left.	<u>Bitwise And:</u> & &.				
	left	<u>Bitwise Or and Exclusive Or:</u> . ^ ^.				
	left	<u>C-style Logical And:</u> &&				
	left	<u>Logical Defined-Or:</u> ^^ //				
	NA	<u>Range Operators:</u>				
	right	<u>Conditional Operator:</u> ? :				
	right	<u>Assignment Operators:</u> =				
		**= += *= &= &.= <<= &&=				
		-= /= = .= >>= =				
		.= %= ^= ^.= //=				
		x=				
		goto last next redo dump				
		, =>				
	left	<u>Comma, fat-comma Operators:</u>				
	NA	<u>list operators (rightward)</u>				
	right	<u>Logical Not:</u> not				
	left	<u>Logical And:</u> and				
	left	<u>Logical or and Exclusive or:</u> or xor				

File test operators	It is possible to combine the file test operator with the AND operator as in the following example:			<pre>if (-e \$fname && -f _ && -r _){ print("\$fname exists and is readable\n"); }</pre>	
The most important operators are shown here. They check if the file...	-r is readable	-e exists.	-b is a block special file.		
	-w is writable	-z is empty.	-c is a character special file.		
	-x is executable	-s has nonzero size (returns size in bytes).	-t handle is opened to a tty.		
	-o is owned by effective uid.	-f is a plain file.	-u has setuid bit set.		
	-R is readable	-d is a directory.	-g has setgid bit set.		
	-W is writable	-l is a symbolic link.	-k has sticky bit set.		
	-X is executable	-p is a named pipe (FIFO) or Filehandle is a pipe.	-T is an ASCII text file (heuristic guess).		
	-O file is owned by real uid.	-S is a socket.	-B is a “binary” file (opposite of -T).		

Perl Special Variables  To get information about a Perl special variable from the command line use the <code>perldoc -v</code> command. <ul style="list-style-type: none">• To get information about <code>\$<</code> use: <code>perldoc -v '\$<'</code>					
• General variables					
default input and pattern searching space	<ul style="list-style-type: none">• <code>\$ARG</code>• <code>\$_</code>		subroutine parameters	<ul style="list-style-type: none">• <code>@ARG</code>• <code>@_</code>	
list separator	<ul style="list-style-type: none">• <code>\$LIST_SEPARATOR</code>• <code>\$"</code>		Subscript separator for multidimensional array emulation	<ul style="list-style-type: none">• <code>\$\$SUBSCRIPT_SEPARATOR</code>• <code>\$\$SUBSEP</code>• <code>\$;</code>	
Name of executed program	<ul style="list-style-type: none">• <code>\$PROGRAM_NAME</code>• <code>\$0</code>		Name used to execute the current copy of Perl	<ul style="list-style-type: none">• <code>\$EXECUTABLE_NAME</code>• <code>\$^X</code>	
Perl process ID	<ul style="list-style-type: none">• <code>\$PROCESS_ID</code>• <code>\$PID</code>• <code>\$\$</code>				
Process real GID	<ul style="list-style-type: none">• <code>\$REAL_GROUP_ID</code>• <code>\$GID</code>• <code>\$(</code>		Process effective GID	<ul style="list-style-type: none">• <code>\$EFFECTIVE_GROUP_ID</code>• <code>\$EGID</code>• <code>\$)</code>	
Process real UID	<ul style="list-style-type: none">• <code>\$REAL_USER_ID</code>• <code>\$UID</code>• <code>\$<</code>		Process effective UID	<ul style="list-style-type: none">• <code>\$EFFECTIVE_USER_ID\$</code>• <code>\$EUID</code>• <code>\$></code>	
Special variables in sort	<ul style="list-style-type: none">• <code>\$a</code>• <code>\$b</code>				
Current environment	<code>%ENV</code> Environment variable accessed as an associative array (a hash). <ul style="list-style-type: none">• See: Perl: How to access shell environment variables through Perl associative arrays.				
Perl interpreter revision, version and subversion	<ul style="list-style-type: none">• <code>\$OLD_PERL_VERSION</code>• <code>\$]</code>		Perl interpreter revision, version and subversion	<ul style="list-style-type: none">• <code>\$PERL_VERSION</code>• <code>\$^V</code>	
Maximum file descriptor	<ul style="list-style-type: none">• <code>\$\$SYSTEM_FD_MAX</code>• <code>\$^F</code>				
Fields of each line when auto-split mode is on.	<code>@F</code>				
Include Directories	<code>@INC</code>	Included filenames	<code>%INC</code>	Hook localization (?)	<code>\$INC</code>

inplace-edit extension value	<ul style="list-style-type: none">\$INPLACE_EDIT\$^I				
Package's class parent classes	@ISA				
Emergency memory pool	\$^M				
Maximum block nesting	\${^MAX_NESTED_EVAL_BEGIN_BLOCKS}				
Name of OS where this Perl was built	<ul style="list-style-type: none">\$OSNAME\$^O				
Signal handlers	%SIG				
Coderefs for various perl keywords	%{^HOOK}				
Time when program began running	<ul style="list-style-type: none">\$BASETIME\$^T				
<ul style="list-style-type: none">Variables related to regular expressions					
captured sub-patterns	\$<digit>(\$1, \$2, ...)				
Capture buffer content	@{^CAPTURE}				
String matched	<ul style="list-style-type: none">\$MATCH\$&	String matched (compiled regexp)	\${^MATCH}		
String preceding match	<ul style="list-style-type: none">\$PREMATCH\$`	String preceding match (compiled regexp)	\${^PREMATCH}		
String following match	<ul style="list-style-type: none">\$POSTMATCH\$'	String following match (compiled regexp)	{^POSTMATCH}		
Last capture group	<ul style="list-style-type: none">\$LAST_PAREN_MATCH\$+	Most recently closed capture group	<ul style="list-style-type: none">\$LAST_SUBMATCH_RESULT\$^N		
Match capture key values	<ul style="list-style-type: none">%{^CAPTURE}%LAST_PAREN_MATCH%+				
Match start offsets	<ul style="list-style-type: none">@LAST_MATCH_START@-	Match ends offsets	<ul style="list-style-type: none">@LAST_MATCH_END@+	Named captured groups	<ul style="list-style-type: none">%{^CAPTURE_ALL}%-
Last successful pattern	\${^LAST_SUCESSFUL_PATTERN}				
Result of last successful regexp assertion	<ul style="list-style-type: none">\$LAST_REGEXP_CODE_RESULT\$^R				
Maximum regexp nested group	\${^RE_COMPILE_RECURSION_LIMIT}				
regexp debug flag	\${^RE_DEBUG_FLAG}				
regexp internal optimization/memory	\${^RE_TRIE_MAXBUF}				
<ul style="list-style-type: none">Variables related to file handles					
Name of current file read from <>	\$ARGV	Command line arguments of the script	@ARGV	Number of arguments minus one	\$#ARGV
Special file handle that iterates over command-line filenames in @ARGV	ARGV	Special file handle that points to currently open output file when doing edit-in-place processing	ARGVOUT		
Output field separator for the print operator	<ul style="list-style-type: none">IO::Handle->output_field_separator(EXPR)\$OUTPUT_FIELD_SEPARATOR\$OFS\$,	Current line number for the last file handled accessed	<ul style="list-style-type: none">HANDLE->input_line_number(EXPR)\$INPUT_LINE_NUMBER\$NR\$.		
Input record separator (newline by default)	<ul style="list-style-type: none">IO::Handle->input_record_separator(EXPR)\$INPUT_RECORD_SEPARATOR\$RS\$/	Output record separator	<ul style="list-style-type: none">IO::Handle->output_record_separator(EXPR)\$OUTPUT_RECORD_SEPARATOR\$ORS\$\		
Auto-flush control	<ul style="list-style-type: none">HANDLE->autoflush(EXPR)\$OUTPUT_AUTOFLUSH\$!	Last read file handle	\${^LAST_FH}		
<ul style="list-style-type: none">Variables related to format					
Current value of the write() accumulator for format() lines.	<ul style="list-style-type: none">\$ACCUMULATOR\$^A				
Form feed format. defaults to \f	<ul style="list-style-type: none">IO::Handle->format_formfeed(EXPR)\$FORMAT_FORMFEED\$^L	Set of characters after which a string may be broken to fill continuation fields	<ul style="list-style-type: none">IO::Handle->format_line_break_characters EXPR\$FORMAT_LINE_BREAK_CHARACTERS\$:		
Number of lines left on the page on currently selected output channel	<ul style="list-style-type: none">HANDLE->format_lines_left(EXPR)\$FORMAT_LINES_LEFT\$-	Current page length of current output channel	<ul style="list-style-type: none">HANDLE->format_lines_per_page(EXPR)\$FORMAT_LINES_PER_PAGE\$=		
Name of current top-page format of output channel	<ul style="list-style-type: none">HANDLE->format_top_name(EXPR)\$FORMAT_TOP_NAME\$^	Report format name of output channel	<ul style="list-style-type: none">HANDLE->format_name(EXPR)\$FORMAT_NAME\$~		
<ul style="list-style-type: none">Error Variables	The variables \$@, \$!, \$^E, and \$? contain information about different types of error conditions that may appear during execution of a Perl program. They correspond to errors detected by the Perl interpreter, C library, operating system, or an external program, respectively.				
Perl error from the last eval operator	<ul style="list-style-type: none">\$EVAL_ERROR\$@	Current state of interpreter	<ul style="list-style-type: none">\$EXCEPTIONS_BEING_CAUGHT\$^S		
Current value of C errno integer variable	<ul style="list-style-type: none">\$OS_ERROR\$ERRNO\$!	Hash of error names to 0 or 1, set to 1 if current error is this error.	<ul style="list-style-type: none">%OS_ERROR%ERRNO%!		
OS detected error	<ul style="list-style-type: none">\$EXTENDED_OS_ERROR\$^E				

Status returned by last pipe close, backtick command, wait, waited, or system() call.	<ul style="list-style-type: none">• \$CHILD_ERROR• \$?	native status returned by last pipe close , backtick command, wait() or wiatpid() or system() call	\${^CHILD_ERROR_NATIVE}
Current value of warning switch	<ul style="list-style-type: none">• \$WARNING• \$^W	Current set of warning checks enabled by the use warnings pragma	\${^WARNING_BITS}
• Variables related to the interpreter state	These variables provide information about the current interpreter state.		
Flag associated with the -c switch	<ul style="list-style-type: none">• \$COMPILING• \$^C	The current value of the debugging flags	<ul style="list-style-type: none">• \$DEBUGGING• \$^D
Current phase of the perl interpreter	\${^GLOBAL_PHASE}		
Compile-time hints for the perl interpreter. Internal use only	\$^H	Values of compiled statements	%^H
Input/Output Layers. Internal use by PerlIO only.	\${^OPEN}		
Debugging support. Internal variable.	<ul style="list-style-type: none">• \$PERLDB• \$^P		
Taint mode	\${^TAINT}	Safe locale operations availability	\${^SAFE_LOCALES}
Unicode Settings of Perl	\${^UNICODE}		
Internal UTF-8 offset caching code state	\${^UTF8CACHE}	State of UTF-8 locale detected by perl at startup.	\${^UTF8LOCALE}
• Deprecated and removed variables:	\$# \$* \$[\${^ENCODING} \${^WIN32_SLOPPY_STAT}		