

Functions		
Arrays	<code>push @arr, \$x</code>	Add to end of array
	<code>\$x = pop @arr;</code>	Remove last element of array, put in \$x
	<code>shift @arr;</code> (See also <code>unshift</code>)	Remove first element of an array, put in \$x
	<code>\$size = scalar @arr;</code>	Number of things in the array
	See also: <code>split</code> , <code>join</code> , <code>splice</code> , <code>sort</code>	<code>split</code> string->array, <code>join</code> array->string, delete part of array, sort array in many ways
scalar variables	<code>while (defined (\$x=<>)) {code}</code>	False if variable has never been set (or when you try to read past the end of an input file)
	<code>length(\$x)</code>	Length of a string
	<code>chomp(\$line);</code>	Remove a newline at the end of a string
	<code>\$short=substr (\$long, 2, 5)</code>	Characters 3-7 of \$long (first char is 0!)
Hashes	<code>@key = keys %hash</code>	The lookup terms in the hash
	<code>if (exists \$hh{"Hsp"}) {...}</code>	See whether hash %hh has a value for key Hsp
Input/Output and Files	<code>open(HANDLE, ">outfile")</code> or <code>die "Can't open \$outfile: \$!\n"</code>	Open outfile for writing, and associate it with filehandle HANDLE. Use "
	<code>print \$x; print HANDLE \$x;</code>	Prints to standard output (screen), Print to filehandle HANDLE
	<code>warn "Something wrong\n";</code>	Prints to standard error (screen)
	<code>\$x=<HANDLE></code>	Read a line from filehandle HANDLE, put in \$x
	<code>close(HANDLE);</code>	Stop reading/writing a previously opened file
Exit	<code>exit;</code>	Exits the program
	<code>die "Something broke!\n";</code>	exits the program with error message

>> Operators and Loops

>> Matching and Regular Expressions

Source :GreyCampus

Training & Certification in Perl :Perl Certification



By **Suha**
cheatography.com/suha/

Published 29th December, 2015.
 Last updated 29th December, 2015.
 Page 1 of 1.

Sponsored by **Readability-Score.com**
 Measure your website readability!
<https://readability-score.com>