

Perl Cheat Sheet

Perl 5 cheat sheet v8 by Juerd Waalboer, <http://juerd.nl/>

```
CONTEXTS      SIGILS                ARRAYS                HASHES
void          $scalar    whole:    @array        %hash
scalar       @array     slice:    @array[0, 2]  @hash{'a',
'b'}
list         %hash      element:  $array[0]     $hash{'a'}
            &sub
            *glob

SCALAR VALUES
number, string, reference, glob,
undef

REFERENCES
\      references          $$foo[1]          aka $foo->[1]
$@%&* dereference        $$foo{bar}        aka $foo->{bar}
[]     anon. arrayref     ${$$foo[1]}[2]    aka $foo->[1]->[2]
{}     anon. hashref      ${$$foo[1]}[2]    aka $foo->[1][2]
\()    list of refs

NUMBERS vs STRINGS  LINKS
OPERATOR PRECEDENCE = =
perl.plover.com
->                +                .
search.cpan.org
++ --            == !=            eq ne
cpan.org
**              < > <= >=    lt gt le ge    pm.org
! ~ \ u+ u-     <=>                cmp            tpj.com
=~ !~

perldoc.com
* / % x          SYNTAX
+ - .           for (LIST) { }, for (a;b;c) {
}
<< >>          while ( ) { }, until ( ) { }
```

```

named uops          if      ( ) { } elsif ( ) { }
else { }
< > <= >= lt gt le ge  unless ( ) { } elsif ( ) { }
else { }
== != <=> eq ne cmp    for equals foreach (ALWAYS)
&
| ^                 REGEX METACHARS           REGEX
MODIFIERS
&&                 ^      string begin           /i case
insens.
||                 $      str. end (before \n) /m line
based ^$
.. ...            +      one or more           /s .
includes \n
?:                *      zero or more           /x ign.
wh.space
= += -= *= etc.  ?      zero or one           /g global
, =>              {3,7} repeat in range
list ops         ()      capture             REGEX
CHARCLASSES
not              (? :) no capture         . == [^\n]
and              []      character class  \s ==
[\x20\f\t\r\n]
or xor           |      alternation         \w == [A-Za-z0-
9_]
                \b      word boundary       \d == [0-9]
                \z      string end           \S, \W and \D
negate
DO
use strict;      DON'T          LINKS
use warnings;    "$foo"         perl.com
my $var;         $$variable_name perlmonks.org
open() or die $!; ` $userinput ` use.perl.org
use Modules;     /$userinput/   perl.apache.org
                parrotcode.org

```

FUNCTION RETURN LISTS				
stat	localtime	caller	SPECIAL VARIABLES	
0 dev	0 second	0 package	\$_	default
variable				
1 ino	1 minute	1 filename	\$0	program name
2 mode	2 hour	2 line	\$/	input
separator				
3 nlink	3 day	3 subroutine	\$\$	output
separator				
4 uid	4 month-1	4 hasargs	\$	autoflush
5 gid	5 year-1900	5 wantarray	\$!	sys/libcall
error				
6 rdev	6 weekday	6 evaltext	\$_	eval error
7 size	7 yearday	7 is_require	\$\$	process ID
8 atime	8 is_dst	8 hints	\$.	line number
9 mtime		9 bitmask	@ARGV	command line
args				
10 ctime	just use		@INC	include
paths				
11 blksize	POSIX::	3..9 only	@_	subroutine
args				
12 blcks	strftime!	with EXPR	%ENV	environment