

Perl Cheatsheet

from Kenward Bradley; updated 24 May 2008

Website: <http://kenwardtown.com>

Comment

```
# don't forget to comment your code!
```

Variables

```
my $scalar = 9 / 18;
my @list = (5, 7, 12, 1);
my $hash{key} = "value";
%hash = %otherhash;
$number += 1; # add
$number -= 1; # subtract
$string .= "append";
```

Print to screen

```
print "string $scalar\n";
```

Print to file

```
open (OUT, '>', $file) || die "Error $!";
print OUT "stuff to print to file\n";
close (OUT) || die "can't close $file:
$!";
# or use '>>' for append
```

Read a file line by line

```
open (INPUT, '<', $file) || die "Error
$!";
while (<INPUT>) {chomp; print $_;}
close(INPUT) || die "can't close $file:
$!";
```

Comparisons

```
if ($string eq "string") {BLOCK;}
else (($this == 1) && ($that > 24))
{BLOCK;}
```

<i>Comparison</i>	<i>Numeric</i>	<i>String</i>
equals	==	eq
not equals	!=	ne
less than	<	lt
greater than	>	gt
less than or equal	<=	le
greater than or equal	>=	ge

<i>Boolean logic</i>	
and	&&
or	
not	!

Loops

```
For ($c=1; $c <= 9; $i++) {BLOCK;}
foreach (@list) {BLOCK;}
foreach (1 .. 9) {BLOCK;}
while ($n eq 32) {BLOCK;}
until ($n eq 32) {BLOCK;}
unless ($n eq 32) {BLOCK;}
```

Subroutine

```
sub routine {
    my ($parama,$paramb) = @_;
    my $result = $parama * $paramb;
    return $result;
}
```

Regular Expressions

Extract using capture

```
$n = "this and that";
$n =~ /^(.*) ( and ) (.*)$/;
# returns $1 = "this", $2 = " and ",
#          $3 = "that"
```

Replace text

```
$c =~ s/this/that/gi;
# Replace "this" with "that" in $c
# g=global, i= case insensitive
```

<i>Symbol</i>	<i>Description</i>
.	Match any character
\d	Digit [0-9]
\s	Whitespace [\ \t\r\n\f]
\w	Alphanumeric [0-9a-zA-Z_]
\n	Linefeed
\\	"\" symbol
\t	Tab
^	At start of string
\$	At end of string
+	Match 1 or more of proceeding
*	Zero or more characters
?	Zero or 1 character
()	Capture
	Or

\D \S \W is negated \d \s \w

Get the time

```
($sec, $min, $hour, $mday, $mon, $year,
$wday, $yday, $isdst) = localtime(time);
```