Perl Cheat Sheet

Perl was originally developed for text manipulation, and now used for tasks including system administration, web development, network programming, GUI development, and more.

Basics

Include libraries with the **use** keyword.

```
use strict;
use warning;
use utf8;
```

Perl modules are available from http://cpan.org

Variables

```
my $var=2;
my $var="foo";
my $var=3.1415;
```

A **scalar** is a single item.

```
my @var=("foo", "bar", 2, 3.1415);
```

An **array** is a list of items.

```
my %var=(
"apple" => "red",
"lime" => "green",
);
```

A **hash** is a collection of key and value pairs.

```
print $var[0];
```

Use **scalar** notation to reference a scalar, even when it's contained in an array or hash.

Functions (subroutines)

```
my $temp=celsius(70);
print "$temp\n";

sub celsius {
   my $f=shift;
   my $cel=($f - 32)*(5/9);
   return($cel);
}
```

Create a subroutine with the keyword **sub**, and place code between braces.

A **return** statement is optional.

Parse parameters with **shift**.

Perl Cheat Sheet

Comparison					
`==` `!=`	equal / not equal	`eq` `ne`	string equal / not equal	`=~` `!~`	string contains / does not contain [*pattern*]
`>` `<`	greater / less than	`gt` `lt`	stringwise greater / less	`<=>` `cmp`	returns -1, 0, or 1 for less, equal, or greater
`>=` `<=`	greater / less than or equal	`ge` `le`	stringwise greater/less than or equal	`//=` `||=`	pair bitwise operators with = when setting variables conditionally

If statement	If (negated)

`if ( condition ) {`
`  # code here`
`} elsif ( other condition ) {`
`  # code here`
`} else {`
`  # code here`
`}`
	`unless ( condition ) {`
	`  # code here`
	`}`

`print "True" if 1 == 1;`	If statements can be used inline
While statement	**While (negated)**

`while ( condition ) {`
`  # code here`
`}`
	`until ( condition ) {`
	`  # code here`
	`}`

`print "Infinite" while 1;`	While statements can be used inline
For each item in array	**For each item in hash**

`foreach (@array) {`
`  print "Item is $_\n";`
`}`
	`foreach my $key (keys %hash) {`
	`  print "Value is $hash{$key}\n";`
	`}`