

```
1: #!/usr/bin/perl
2: # $Id: pkeyvalue.perl,v 1.1 2019-04-24 17:09:28-07 - - $
3: #
4: # Documentation:  see assignment specifications.
5: #
6: use strict;
7: use warnings;
8:
9: $0 =~ s|^(\./)([^\./]+)/*$|$2|;
10: my $exit_status = 0;
11: END {exit $exit_status}
12: $SIG{'__WARN__'} = sub {print STDERR "$0: @_"; $exit_status = 1};
13: $SIG{'__DIE__'} = sub {warn @_; exit};
14:
15: my %map;
16:
17: sub printkeyval ($) {
18:     my ($key) = @_;
19:     my $val = $map{$key};
20:     print $key, defined $val ? " = $val\n" : ": key not found\n";
21: }
22:
23: for my $filename (@ARGV ? @ARGV : "-") {
24:     open my $file, "<$filename" or do {warn "$filename: $!\n"; next};
25:     while (defined (my $line = <$file>)) {
26:         chomp $line;
27:         printf "%s: %d: %s\n", $filename, $., $line;
28:         next if $line =~ m/^\s*(#|$)/;
29:         my ($key, $val) = map {m/^\s*(.*?)\s*/g} split m/=/, $line, 2;
30:         unless (defined $val) {
31:             printkeyval $key;
32:         } elsif (length $key) {
33:             if ($key eq "/" ) {
34:                 map {printkeyval $_ unless length $val and $map{$_} ne $val}
35:                     sort {$b cmp $a} keys %map; }
36:             elsif (length $val) {
37:                 $map{$key} = $val; printkeyval $key;
38:             } else {
39:                 delete $map{$key}
40:             }
41:         } else {
42:             map {printkeyval $_ unless length $val and $map{$_} ne $val}
43:                 sort keys %map; }
44:     }
45: }
46:
```