A leading colon removes the 'name=' part of the response, this allows you to map to the name you need.

```
$ echo "goodvfork="`./perl -Ilib -V::usevfork`
goodvfork=false;
```

Leading and trailing colons can be used together if you need positional parameter values without the names. Note that in the case below, the PERL_API params are returned in alphabetical order.

```
$ echo building_on `perl -V::osname: -V::PERL_API_.*:` now
building_on 'linux' '5' '1' '9' now
```

-w

prints warnings about dubious constructs, such as variable names that are mentioned only once and scalar variables that are used before being set, redefined subroutines, references to undefined filehandles or filehandles opened read-only that you are attempting to write on, values used as a number that doesn't look like numbers, using an array as though it were a scalar, if your subroutines recurse more than 100 deep, and innumerable other things.

This switch really just enables the internal `$^W` variable. You can disable or promote into fatal errors specific warnings using `__WARN__` hooks, as described in *perlvar* and warn in *perlfunc*. See also *perldiag* and *perltrap*. A new, fine-grained warning facility is also available if you want to manipulate entire classes of warnings; see *warnings* or *perllexwarn*.

-W

Enables all warnings regardless of `no warnings` or `$^W`. See *perllexwarn*.

-X

Disables all warnings regardless of `use warnings` or `$^W`. See *perllexwarn*.

-x

-x *directory*

tells Perl that the program is embedded in a larger chunk of unrelated ASCII text, such as in a mail message. Leading garbage will be discarded until the first line that starts with #! and contains the string "perl". Any meaningful switches on that line will be applied. If a directory name is specified, Perl will switch to that directory before running the program. The **-x** switch controls only the disposal of leading garbage. The program must be terminated with `__END__` if there is trailing garbage to be ignored (the program can process any or all of the trailing garbage via the DATA filehandle if desired).

35.3 ENVIRONMENT

HOME

Used if chdir has no argument.

LOGDIR

Used if chdir has no argument and HOME is not set.

PATH

Used in executing subprocesses, and in finding the program if **-S** is used.

PERL5LIB

A list of directories in which to look for Perl library files before looking in the standard library and the current directory. Any architecture-specific directories under the specified locations are automatically included if they exist. If PERL5LIB is not defined, PERLLIB is used. Directories are separated (like in PATH) by a colon on unixish platforms and by a semicolon on Windows (the proper path separator being given by the command `perl -V:path_sep`).

When running taint checks (either because the program was running setuid or setgid, or the **-T** switch was used), neither variable is used. The program should instead say: