



Full credit is given to the above companies including the Operating System (OS) that this PDF file was generated!

Rocky Enterprise Linux 9.2 Manual Pages on command 'namespace::autoclean.3pm'

\$ man namespace::autoclean.3pm

namespace::autoclean(3pm) User Contributed Perl Documentation namespace::autoclean(3pm)

NAME

namespace::autoclean - Keep imports out of your namespace

VERSION

version 0.29

SYNOPSIS

```
package Foo;

use namespace::autoclean;

use Some::Package qw/imported_function/;

sub bar { imported_function('stuff') }

# later on:

Foo->bar;          # works

Foo->imported_function; # will fail. imported_function got cleaned after compilation
```

DESCRIPTION

When you import a function into a Perl package, it will naturally also be available as a method.

The "namespace::autoclean" pragma will remove all imported symbols at the end of the current package's compile cycle. Functions called in the package itself will still be bound by their name, but they won't show up as methods on your class or instances.

This module is very similar to namespace::clean, except it will clean all imported functions, no matter if you imported them before or after you "use"d the pragma. It will also not touch anything that looks like a method.

If you're writing an exporter and you want to clean up after yourself (and your peers),

you can use the "-cleanee" switch to specify what package to clean:

```
package My::MooseX::namespace::autoclean;

use strict;

use namespace::autoclean (); # no cleanup, just load

sub import {

    namespace::autoclean->import(

        -cleanee => scalar(caller),

    );

}
```

WHAT IS AND ISN'T CLEANED

"namespace::autoclean" will leave behind anything that it deems a method. For Moose classes, this is based on the "get_method_list" method from the Class::MOP::Class. For non-Moose classes, anything defined within the package will be identified as a method. This should match Moose's definition of a method. Additionally, the magic subs installed by overload will not be cleaned.

PARAMETERS

-also => [ITEM | REGEX | SUB, ..]

-also => ITEM

-also => REGEX

-also => SUB

Sometimes you don't want to clean imports only, but also helper functions you're using in your methods. The "-also" switch can be used to declare a list of functions that should be removed additional to any imports:

```
use namespace::autoclean -also => ['some_function', 'another_function'];
```

If only one function needs to be additionally cleaned the "-also" switch also accepts a plain string:

```
use namespace::autoclean -also => 'some_function';
```

In some situations, you may wish for a more powerful cleaning solution.

The "-also" switch can take a Regex or a CodeRef to match against local function names to clean.

```
use namespace::autoclean -also => qr/^_/

use namespace::autoclean -also => sub { $_ =~ m{^_} };

use namespace::autoclean -also => [qr/^_/ , qr/^hidden_/];
```

```
use namespace::autoclean -also => [sub { $_ =~ m/^_/ or $_ =~ m/^hidden/ }, sub { uc($_) == $_ }];
```

```
-except => [ ITEM | REGEX | SUB, .. ]
```

```
-except => ITEM
```

```
-except => REGEX
```

```
-except => SUB
```

This takes exactly the same options as "-also" except that anything this matches will not be cleaned.

CAVEATS

When used with Moo classes, the heuristic used to check for methods won't work correctly for methods from roles consumed at compile time.

```
package My::Class;

use Moo;

use namespace::autoclean;

# Bad, any consumed methods will be cleaned

BEGIN { with 'Some::Role' }

# Good, methods from role will be maintained

with 'Some::Role';
```

Additionally, method detection may not work properly in Mouse classes in perls earlier than 5.10.

SEE ALSO

- ? namespace::clean
- ? B::Hooks::EndOfScope
- ? namespace::sweep
- ? Sub::Exporter::ForMethods
- ? Sub::Name
- ? Sub::Install
- ? Test::CleanNamespaces
- ? Dist::Zilla::Plugin::Test::CleanNamespaces

SUPPORT

Bugs may be submitted through the RT bug tracker

<<https://rt.cpan.org/Public/Dist/Display.html?Name=namespace-autoclean>> (or

bug-namespace-autoclean@rt.cpan.org <<mailto:bug-namespace-autoclean@rt.cpan.org>>).

There is also a mailing list available for users of this distribution, at

<<http://lists.perl.org/list/moose.html>>.

There is also an irc channel available for users of this distribution, at "#moose" on "irc.perl.org" <<irc://irc.perl.org/#moose>>.

AUTHOR

Florian Ragwitz <rafl@debian.org>

CONTRIBUTORS

- ? Karen Etheridge <ether@cpan.org>
- ? Graham Knop <haarg@haarg.org>
- ? Dave Rolsky <autarch@urth.org>
- ? Kent Fredric <kentfredric@gmail.com>
- ? Tomas Doran <bobtfish@bobtfish.net>
- ? Shawn M Moore <cpan@sartak.org>
- ? Felix Ostmann <sadrak@cpan.org>
- ? Chris Prather <chris@prather.org>
- ? Andrew Rodland <arodland@cpan.org>

COPYRIGHT AND LICENCE

This software is copyright (c) 2009 by Florian Ragwitz.

This is free software; you can redistribute it and/or modify it under the same terms as the Perl 5 programming language system itself.

perl v5.30.0

2019-10-13

namespace::autoclean(3pm)