

**NAME**

MooX::Aliases – easy aliasing of methods and attributes in Moo

**SYNOPSIS**

```
package MyClass;
use Moo;
use MooX::Aliases;

has this => (
    is      => 'rw',
    alias => 'that',
);

sub foo { my $self = shift; print $self->that }
alias bar => 'foo';

my $o = MyApp->new();
$o->this('Hello World');
$o->bar; # prints 'Hello World'

or

package MyRole;
use Moo::Role;
use MooX::Aliases;

has this => (
    is      => 'rw',
    alias => 'that',
);

sub foo { my $self = shift; print $self->that }
alias bar => 'foo';
```

**DESCRIPTION**

The MooX::Aliases module will allow you to quickly alias methods in Moo. It provides an alias parameter for *has()* to generate aliased accessors as well as the standard ones. Attributes can also be initialized in the constructor via their aliased names.

You can create more than one alias at once by passing a listref:

```
has ip_addr => (
    alias => [ qw(ipAddr ip) ],
);
```

**FUNCTIONS**

alias \$alias, \$method  
Creates \$alias as a method that is aliased to \$method.

**CAVEATS**

This module uses the BUILDARGS to map the attributes. If a class uses a custom BUILDARGS, this module may not behave properly.

**SEE ALSO**

MooseX::Aliases

**AUTHOR**

haarg – Graham Knop (cpan:HAARG) <haarg@haarg.org>

**CONTRIBUTORS**

- Chris Prather <chris@prather.org>
- Jesse Luehrs <doy@tozt.net>
- Justin Hunter <justin.d.hunter@gmail.com>
- Karen Etheridge <ether@cpan.org>
- Yuval Kogman <nothingmuch@woobling.org>
- Daniel Gempesaw <gempesaw@gmail.com>
- Denis Ibaev <dionys@gmail.com>

**COPYRIGHT**

Copyright (c) 2013 the MooX::Alises “AUTHOR” and “CONTRIBUTORS” as listed above.

**LICENSE**

This library is free software and may be distributed under the same terms as perl itself.