



Linux Ubuntu 22.4.5 Manual Pages on command 'File::Find::Rule::Procedural.3pm'

\$ man File::Find::Rule::Procedural.3pm

File::Find::Rule::ProceduralUser)Contributed Perl DocumenFile::Find::Rule::Procedural(3pm)

NAME

File::Find::Rule::Procedural - File::Find::Rule's procedural interface

SYNOPSIS

```
use File::Find::Rule;
```

```
# find all .pm files, procedurally
```

```
my @files = find(file => name => '*.pm', in => \@INC);
```

DESCRIPTION

In addition to the regular object-oriented interface, File::Find::Rule provides two subroutines for you to use.

```
"find( @clauses )"
```

```
"rule( @clauses )"
```

"find" and "rule" can be used to invoke any methods available to the OO version. "rule" is a synonym for "find"

Passing more than one value to a clause is done with an anonymous array:

```
my $finder = find( name => [ '*.mp3', '*.ogg' ] );
```

"find" and "rule" both return a File::Find::Rule instance, unless one of the arguments is "in", in which case it returns a list of things that match the rule.

```
my @files = find( name => [ '*.mp3', '*.ogg' ], in => $ENV{HOME} );
```

Please note that "in" will be the last clause evaluated, and so this code will search for mp3s regardless of size.

```
my @files = find( name => '*.mp3', in => $ENV{HOME}, size => '<2k' );
```

^

|

Clause processing stopped here -----/

It is also possible to invert a single rule by prefixing it with "!" like so:

```
# large files that aren't videos
```

```
my @files = find( file =>
    '!name' => [ '*.avi', '*.mov' ],
    size   => '>20M',
    in     => $ENV{HOME} );
```

AUTHOR

Richard Clamp <richardc@unixbeard.net>

COPYRIGHT

Copyright (C) 2003 Richard Clamp. All Rights Reserved.

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

SEE ALSO

File::Find::Rule

perl v5.20.2

2015-12-03

File::Find::Rule::Procedural(3pm)