

NAME

TAP::Parser::YAMLish::Writer - Write YAMLish data

VERSION

Version 3.36

SYNOPSIS

```

use TAP::Parser::YAMLish::Writer;

my $data = {
    one => 1,
    two => 2,
    three => [ 1, 2, 3 ],
};

my $yw = TAP::Parser::YAMLish::Writer->new;

# Write to an array...
$yw->write( $data, \@some_array );

# ...an open file handle...
$yw->write( $data, $some_file_handle );

# ...a string ...
$yw->write( $data, \$some_string );

# ...or a closure
$yw->write( $data, sub {
    my $line = shift;
    print "$line\n";
} );

```

DESCRIPTION

Encodes a scalar, hash reference or array reference as YAMLish.

METHODS**Class Methods***new*

```
my $writer = TAP::Parser::YAMLish::Writer->new;
```

The constructor `new` creates and returns an empty `TAP::Parser::YAMLish::Writer` object.

Instance Methods*write*

```
$writer->write($obj, $output );
```

Encode a scalar, hash reference or array reference as YAML.

```

my $writer = sub {
    my $line = shift;
    print SOMEFILE "$line\n";
};

my $data = {
    one => 1,
    two => 2,
    three => [ 1, 2, 3 ],
};

```

```
my $yw = TAP::Parser::YAMLish::Writer->new;  
$yw->write( $data, $writer );
```

The `$output` argument may be:

- a reference to a scalar to append YAML to
- the handle of an open file
- a reference to an array into which YAML will be pushed
- a code reference

If you supply a code reference the subroutine will be called once for each line of output with the line as its only argument. Passed lines will have no trailing newline.

AUTHOR

Andy Armstrong, <andy@hexten.net>

SEE ALSO

YAML::Tiny, YAML, YAML::Syck, Config::Tiny, CSS::Tiny, <<http://use.perl.org/~Alias/journal/29427>>

COPYRIGHT

Copyright 2007-2011 Andy Armstrong.

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

The full text of the license can be found in the LICENSE file included with this module.