



**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 'Test2::EventFacet::Info.3perl'***

***\$ man Test2::EventFacet::Info.3perl***

Test2::EventFacet::Info(3perl) Perl Programmers Reference Guide Test2::EventFacet::Info(3perl)

NAME

Test2::EventFacet::Info - Facet for information a developer might care about.

DESCRIPTION

This facet represents messages intended for humans that will help them either understand a result, or diagnose a failure.

NOTES

This facet appears in a list instead of being a single item.

FIELDS

`$string_or_structure = $info->{details}`

`$string_or_structure = $info->details()`

Human readable string or data structure, this is the information to display.

Formatters are free to render the structures however they please. This may contain a blessed object.

If the "table" attribute (see below) is set then a renderer may choose to display the table instead of the details.

`$structure = $info->{table}`

`$structure = $info->table()`

If the data the "info" facet needs to convey can be represented as a table then the data may be placed in this attribute in a more raw form for better display. The data must also be represented in the "details" attribute for renderers which do not support rendering tables directly.

The table structure:

```

my %table = {
    header => [ 'column 1 header', 'column 2 header', ... ], # Optional
    rows => [
        ['row 1 column 1', 'row 1, column 2', ... ],
        ['row 2 column 1', 'row 2, column 2', ... ],
        ...
    ],
    # Allow the renderer to hide empty columns when true, Optional
    collapse => $BOOL,
    # List by name or number columns that should never be collapsed
    no_collapse => \@LIST,
}

```

```
$short_string = $info->{tag}
```

```
$short_string = $info->tag()
```

Short tag to categorize the info. This is usually 10 characters or less, formatters may truncate longer tags.

```
$bool = $info->{debug}
```

```
$bool = $info->debug()
```

Set this to true if the message is critical, or explains a failure. This is info that should be displayed by formatters even in less-verbose modes.

When false the information is not considered critical and may not be rendered in less-verbose modes.

```
$bool = $info->{important}
```

```
$bool = $info->important
```

This should be set for non debug messages that are still important enough to show when a formatter is in quiet mode. A formatter should send these to STDOUT not STDERR, but should show them even in non-verbose mode.

## SOURCE

The source code repository for Test2 can be found at <http://github.com/Test-More/test-more/>.

## MAINTAINERS

Chad Granum <[exodist@cpan.org](mailto:exodist@cpan.org)>

## AUTHORS

Chad Granum <exodist@cpan.org>

## COPYRIGHT

Copyright 2020 Chad Granum <exodist@cpan.org>.

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

See <http://dev.perl.org/licenses/>

perl v5.34.0

2023-11-23

Test2::EventFacet::Info(3perl)